ATARI ST

Lüers
Stein

ATARI ST für Einsteiger

DATA BECKER

Lüers
Stein

Atari ST für Einsteiger

DATA BECKER

ISBN 3-89011-152-1

4. überarbeitete und erweiterte Auflage

Wichtiger Hinweis:
Die in diesem Buch wiedergegebenen Schaltungen, Verfahren und Programme werden ohne Rücksicht auf die Patentlage mitgeteilt. Sie sind ausschließlich für Amateur- und Lehrzwecke bestimmt und dürfen nicht gewerblich genutzt werden.

Alle Schaltungen, technischen Angaben und Programme in diesem Buch wurden von dem Autoren mit größter Sorgfalt erarbeitet bzw. zusammengestellt und unter Einschaltung wirksamer Kontrollmaßnahmen reproduziert. Trotzdem sind Fehler nicht ganz auszuschließen. DATA BECKER sieht sich deshalb gezwungen, darauf hinzuweisen, daß weder eine Garantie noch die juristische Verantwortung oder irgendeine Haftung für Folgen, die auf fehlerhafte Angaben zurückgehen, übernommen werden kann. Für die Mitteilung eventueller Fehler ist der Autor jederzeit dankbar.

Danke!

Bei der Arbeit für dieses Buch berieten uns ATARI ST-Freaks aus vielen Orten in Deutschland, namentlich genannt sein sollen an dieser Stelle:

Julian Reschke
Harald Mieling
Ludger Pohaska

Für das äußere Erscheinungsbild (Karikaturen) in diesem Buch zeichnen verantwortlich:

Daniel Stoffregen
Olaf Schellenberger
Ronald Goergen

Vielen Dank auch der Redaktion der DATA WELT, die uns großzügig mit Informationen und Fotomaterial versorgte. Die Fotos machten:

Cäcilia Jordan
Helmut Kaczmarek
Matthias Ziert

Münster und Bochum, im Juli 1986

Rainer Lüers und Michael Stein

Inhaltsverzeichnis

Liebe Kolleginnen und Kollegen!

Wir dürfen Sie doch so nennen, oder? Immerhin sind Sie auf dem besten Wege, ein echter Computerprofi zu werden. Sie zweifeln daran? Wir sind uns da ganz sicher!

Schließlich lesen Sie gerade in einem Buch, das die Absicht hat, Sie mit einem der leistungsfähigsten und schnellsten Rechner des derzeitigen Marktes vertraut zu machen. Wenn das kein Grund ist, Sie als angehenden Profi zu bezeichnen! Sie werden schon sehen, es wird nicht lange dauern, und Sie gehen mit Ihrem ATARI ST um, als hätten Sie Ihr Leben lang nichts anderes getan.

Aus fünf Kapiteln besteht das Buch, wobei Sie nach den zwei Startkapiteln, wo wir zusammen erste Kontakte zu unserem ATARI ST knüpfen wollen, im dritten Kapitel dann in die Programmiersprache BASIC eintauchen werden. Sind Sie dann noch informationshungrig, haben wir im vierten Kapitel Interessantes rund um den ATARI ST für Sie zusammengestellt.

Nach der Lektüre dieses Buches werden Sie also Ihren Computer kennen, die komfortablen Bedienungsmöglichkeiten beherrschen und Ihren Rechner selbst in BASIC programmieren können. Bitte erwarten Sie keine ausführliche, wissenschaftliche Abhandlung über den ATARI ST von uns. Wir wollen versuchen, mit Ihnen zusammen so angenehm wie möglich diesen Supercomputer zu ergründen. Vielleicht bekommen Sie ja dann auch Appetit, sich über das eine oder andere Thema genauer zu informieren. Sie dorthin zu führen, das haben wir uns vorgenommen.

Sind Sie bereit? Dann genug der salbungsvollen Worte und auf ins Abenteuer ATARI ST.

1. Die ersten Schritte: Aufstellen und Anschließen

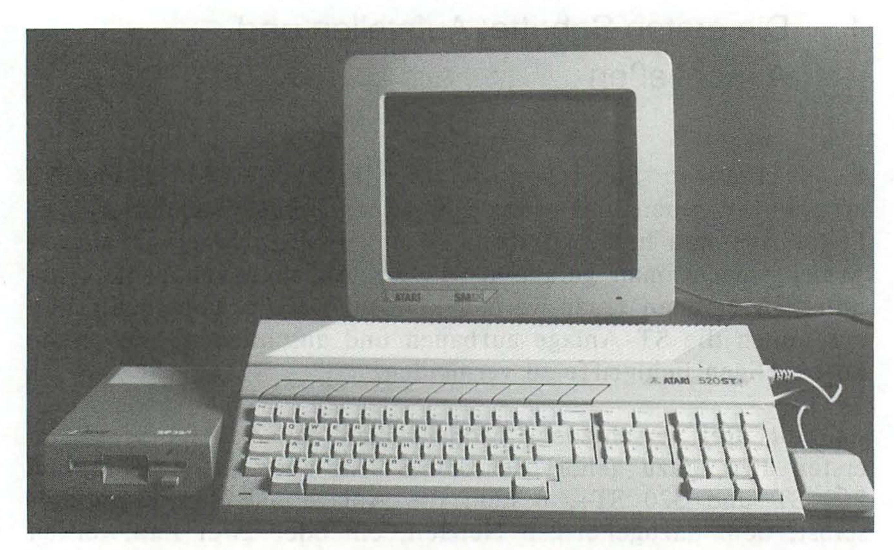
Sicher brennen Sie schon darauf, Ihren ATARI ST endlich arbeiten zu sehen und etwas von seinen herausragenden Fähigkeiten aus ihm herauszukitzeln. Wie wir aus eigener Erfahrung wissen, macht man gerade in diesem Anfangseifer Fehler, die unter Umständen teuer werden können. Darum wollen wir nun zusammen die ST-Anlage aufbauen und anschließen, um solche kostspieligen Fehlgriffe zu vermeiden.

Es gibt drei verschiedene Modellreihen des ATARI ST. Beim ersten Typ, dazu gehört der 520 ST/M, aber auch die älteren 260 ST und 520 ST+, besteht die Anlage aus dem Computer selbst, dem dazugehörigen Netzteil, ein oder zwei Laufwerken und der "Maus". Bei den Typen 260 ST und 520 ST+ kommt in jedem Fall noch ein Monitor dazu, den Sie auch an den 520 ST/M anschließen können.

Die zweite Art, den ST zu erleben, ist der 1040 STF. Bei diesem Rechner wurde nicht nur das Diskettenlaufwerk in das Computergehäuse eingebaut, auch Netzteile braucht man beim "großen" ST nicht, weil ein sehr leistungsfähiges Netzteil ebenfalls im Computer selbst steckt. Zum Arbeiten ist dann eigentlich nur noch ein Monitor und - natürlich - die Maus nötig.

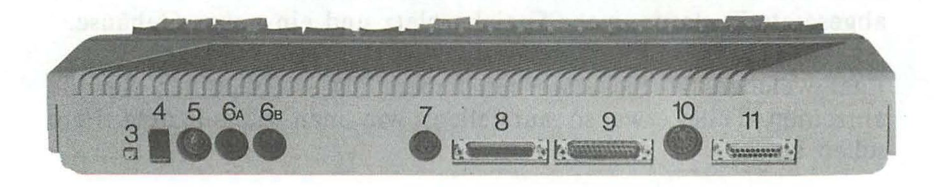
Das neueste Produkt in der ST-Palette ist der Mega ST. Man bekommt ihn als Mega ST 2 und Mega ST 4, in dem wie beim 1040 STF das Diskettenlaufwerk und die Netzteile eingebaut sind. Sie unterscheiden sich von den restlichen STs durch eine abgesetzte Tastatur, mehr Speicherplatz und ein neues Gehäuse.

Egal welchen ST Sie erworben haben, als erstes sollten Sie die einzelnen Teile etwa so aufstellen, wie man es auf dem Foto sehen kann.



Nun wollen wir die Anlage anschließen und die einzelnen Geräte miteinander verbinden. Weil die Modelltypen 520 ST/M (260 ST, 520 ST+), 1040 STF und Mega ST doch etwas unterschiedlich anzuschließen sind, finden Sie im folgendem immer den Hinweis darauf, für welchen ST nun was gilt. Tun Sie sich und uns einen Gefallen? Wenn wir nun gemeinsam die Computeranlage aufbauen, so achten Sie bitte darauf, die Netzstecker der Netzteile erst ganz zum Schluß in die Steckdose zu stecken. Beginnen wir damit, wie beim 520 ST/M (260 ST, 520 ST+) Rechner, Floppy und Monitor bzw. Fernseher verbunden werden.

1.1 Der Computer

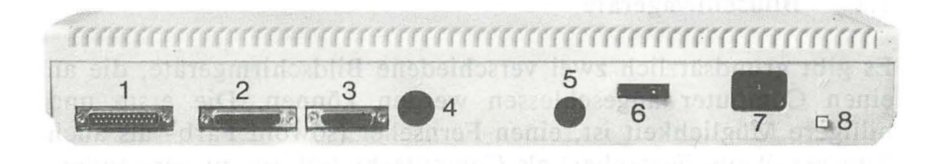


Sehen wir uns dazu zunächst den ATARI ST/M (260 ST, 520 ST+) von hinten an. Die drei Geräte unterscheiden sich in der Lage der Anschlüsse nicht voneinander. Deutlich kann man die

Anschlüsse erkennen. Damit wir sie besser unterscheiden können, haben wir sie durchnumeriert.

Als Besitzer eines ATARI 520 ST/M werden Sie bemerkt haben, das sich bei Ihrem Rechner zwischen Buchse 6 B und Buchse 7 eine weitere Buchse befindet. Diese Buchse dient zum Anschluß Ihres Computers an einen Fernseher über ein Antennenkabel, daß im Lieferumfang enthalten ist. Das "M" in der Bezeichnung Ihres Computers steht stellvertretend für den eingebauten HF-Modulator, der in der Lage ist, das Computerbild auf einem normalen Fernseher darzustellen. Zum Anschluß lesen Sie bitte Kapitel 1.2.2.

Zuerst wollen wir das Netzteil des ST's miteinander verbinden. Dafür stecken Sie den runden Stecker des Netzteils in Buchse 5 am ATARI ST. Achten Sie darauf, daß die kleine Kerbe am Netzteilstecker nach oben zeigt, wenn Sie ihn in den ST stecken. Beim 1040 STF sieht alles etwas leichter aus. So sieht er von hinten aus:



Das Stromkabel, mit dem der 1040 STF seine Energie aus dem Stromnetz saugt, wird in die Buchse 7 am Rechner gesteckt. Das ist schon fast alles.

Auch beim Mega ST wird für die Stromversorgung lediglich das Stromkabel in Buchse 7 gesteckt.



1.2 Bildschirmgeräte

Es gibt grundsätzlich zwei verschiedene Bildschirmgeräte, die an einen Computer angeschlossen werden können. Die erste und billigere Möglichkeit ist, einen Fernseher (sowohl Farb- als auch Schwarz-Weiß-Fernseher) als Computerbildschirm zu verwenden. Diese Möglichkeit können Sie jedoch nur mit einem 520 ST/M nutzen. Die zweite Art von Bildschirmgeräten nennt man Monitore. Sicher haben Sie das gestochen scharfe Bild des Schwarz-Weiß-Monitors SM 124 bei Ihrem Händler bewundert.

Grundsätzlich ist ein Arbeiten mit beiden Geräten möglich, jedoch ist ein Monitor aufgrund der besseren Darstellungsmöglichkeit für professionelle Anwendungen zu empfehlen.

1.2.1 Der Monitor

Das Netzkabel des Monitors muß hinten rechts in den Monitor gesteckt werden. Das bereits fest am Monitor hängende Kabel gehört nun in die Buchse 7 des 260 STs. Beim 1040 ST und Mega ST trägt die Buchse die Bezeichnung 5. Auch hier muß die kleine Kerbe wieder nach oben zeigen.

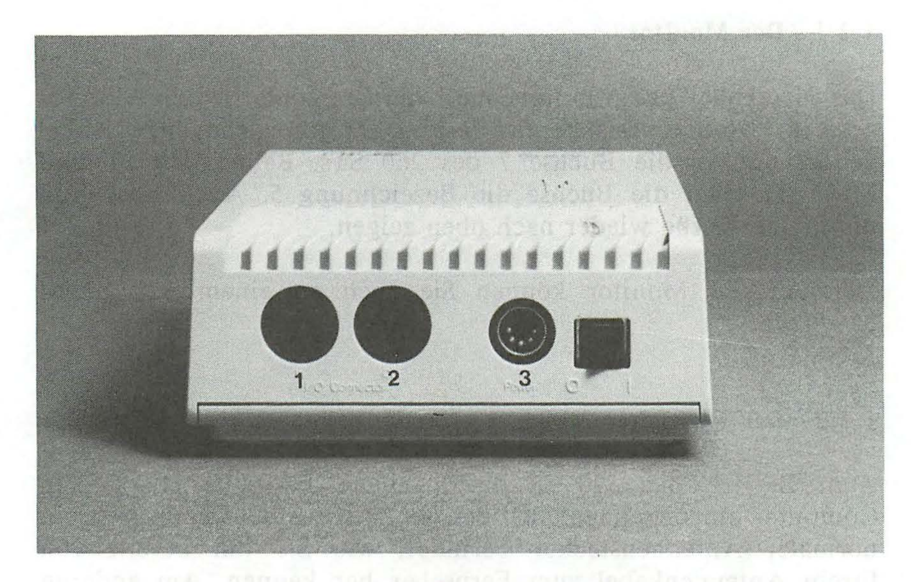
Den ATARI Monitor können Sie auch an einem 520 ST/M betreiben.

1.2.2 Der Fernseher

Zum Betrieb des 520 ST/M an einem Fernseher liegt dem Computer ein Antennenkabel bei, an dessen einem Ende sich ein normaler Antennenstecker befindet, wie Sie ihn bereits von Ihrem Antennenkabel zum Fernseher her kennen. Am anderen Ende befindet sich ebenfalls ein runder Stecker, dessen Mittelstift jedoch hervorschaut und etwas dicker ist. Dieser Stecker wird Cinch-Stecker genannt und wird in Ihren Computer eingesteckt, und zwar in die Buchse zwischen Buchse 6 B und 7. Den Antennenstecker dieser Leitung stecken Sie bitte in Ihr Fernsehgerät in die Antennenbuchse.

1.3 Die Diskettenstation

Die Diskettenstation, oder auch Floppy genannt, muß, wie wir es gerade beim 260 ST und 520 ST+ gemacht haben, mit Ihrem Netzteil verbunden werden. Der Netzteilstecker gehört, wieder mit der Kerbe nach oben, in den Anschluß 3 der Diskettenstation. Für 1040 STF- und Mega ST-Besitzer gilt dies natürlich nur, wenn Sie sich neben dem eingebauten Laufwerk noch ein zweites leisten konnten.



Nun braucht die Floppy noch Kontakt zu ihrem Chef, dem ST. Zu diesem Zweck steckt man das graue Kabel, das im Paket der Floppy lag, in den Anschluß 1 der Floppy. Das andere Ende kommt in Anschluß 10 des Computers. Besitzer eines 1040 STF oder eines Mega ST sind nun schon fertig und können sich direkt dem Kapitel 1.1.4. widmen.

Sollten Sie als Besitzer eines 520 ST/M (260ST, 520ST+) tatsächlich zwei Floppies Ihr eigen nennen, verfahren Sie zunächst mit der ersten so wie gerade beschrieben. Dann verbinden Sie auch die zweite Floppy mit ihrem Netzteil. Das der zweiten Floppy beiliegende graue Kabel stecken Sie nun in Anschluß 2 der ersten und in Anschluß 1 der zweiten Diskettenstation.

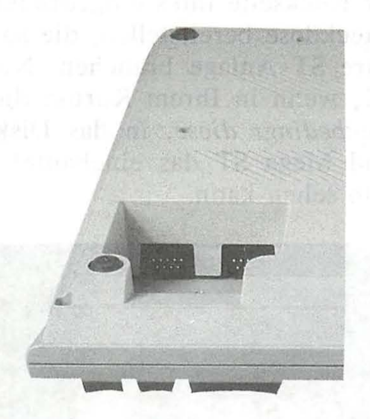
1.4 Die Tastatur

Dieses Kapitel gilt nur für Mega ST-Besitzer, da bei den anderen STs die Tastatur fester Bestandteil des Rechners und intern verbunden ist. Das Spiralkabel wird links am Rechner in die entsprechende Buchse gesteckt. An der Tastatur wird der Stecker an der Rückseite zwischen den beiden Aussparungen für Maus und Joysticks eingesteckt. Falls Sie die Verbindung einmal lösen wollen, so drücken Sie die Sicherungslasche gegen den Stecker und ziehen ihn vorsichtig heraus.

1.5 Die Maus

Bis zum Start mit Ihrer kompletten Anlage ist es nun nur noch ein kleiner Schritt. Nun muß nur noch die Maus angeschlossen werden. Beim 520 ST/M, 260 ST und 520ST+ finden Sie an der rechten Seite zwei Anschlüsse. An den mit "0" gekennzeichneten der beiden schließen Sie nun die Maus an, sofern vorhanden.

1040-ST-Benutzer müssen, um die Maus anzuschließen, den Computer etwas anheben und finden unter dem Gerät auf der rechten Seite einen kleinen Schacht, der die Mausanschlüsse verbirgt.

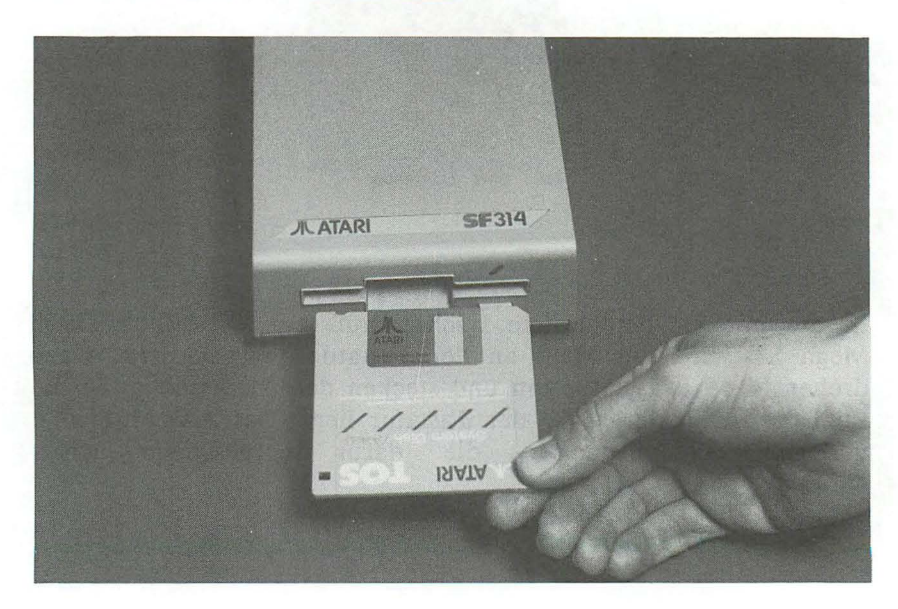


In die mit "0" beschriftete Buchse muß nun die Maus. Beim Mega ST wird die Maus an der Tastatur angeschlossen. Dazu drehen Sie die Tastatur um und stecken den Mausstecker in die Buchse, die in der Mitte der Tastatur liegt. Das Anschlußkabel verlegen Sie bitte in die dafür vorgesehene Rille,

damit die Tastatur später gerade auf dem Tisch aufliegt. Fertig! Da Sie die Maus zur Bedienung Ihres ATARI ST eigentlich ständig gebrauchen, ist es ratsam, sie direkt auf der rechten Seite liegen zu lassen.

1.6 Startvorbereitungen

Bevor wir nun die Stecker der Netzteile (bzw. des STs selbst) und des Monitors bzw. Fernsehers mit dem Stromnetz verbinden, vergewissern Sie sich, daß alle Geräte ausgeschaltet sind. Beim ST ist der Schalter an der Rückseite (siehe Foto), beim Monitor ist das Rad "ON/VOLUME" so lange zu drehen, bis es einrastet (nur bei Original Atari-Monitoren), und die Floppy gehorcht dem Schalter hinten. Das Floppylaufwerk ist an, wenn der Kippschalter an der Rückseite links eingedrückt ist. Jetzt sollten Sie eine Mehrfachsteckdose bereitstellen, die so viele Anschlüsse hat, wie Sie für Ihre ST-Anlage brauchen. Nun sollten Sie die LANGUAGE DISK, wenn in Ihrem Karton die SYSTEM DISK dabei war jedoch *unbedingt diese*, in das Diskettenlaufwerk A (beim 1040 STF und Mega ST das einebaute) einlegen, so wie man es auf dem Foto sehen kann.



Jetzt können wir endlich die Verbindung zum Stromnetz herstellen, und die Geräte anschließend in folgender Reihenfolge einschalten:

1. Monitor bzw. Fernseher
2. Diskettenstation 2 (falls vorhanden)
3. Diskettenstation 1 (falls vorhanden)
4. Computer

1.5.1 Kanaleinstellung bei Fernsehbetrieb

Wenn Sie alle Geräte eingeschaltet haben, so müssen Sie, für den Fall, daß Sie Ihren 520 ST/M an ein Fernsehgerät angeschlossen haben, Ihren Fernseher auf den richtigen Kanal einstellen. Dies geht genauso, als ob Sie ein Fernsehprogramm einstellen wollen.

Da die Einstellungen von Gerät zu Gerät etwas abweichen können, bitten wir Sie, sich die Bedienungsanleitung Ihres Fernsehers hervorzuholen.

Es gibt grundsätzlich 3 verschiedene Möglichkeiten, wie ein Fernseher auf den richtigen Kanal eingestellt wird. Dies ist von dem Modell des Fernsehers abhängig. Zuerst suchen wir uns einen freien Programmspeicher aus, z.B. Nummer 8. Nummer 1 bis Nummer 3 sind im allgemeinen schon mit den Fernsehprogrammen eingestellt, die Sie unverändert lassen sollten, um mit einfachem Antennensteckerwechsel zwischen Fernseh- und Computerbetrieb hin- und herschalten zu können (sonst artet Ihr gemütlicher Fernsehabend in Programmsucherei aus).

1. Möglichkeit: Sendersuchlauf

Stellen Sie Ihren Fernseher um auf den UHF-Bereich, der meist auch nur mit einem großen U gekennzeichnet ist. Betätigen Sie nun die Sendersuchlauf-Taste so oft, bis Sie das Computerbild sehen. Speichern Sie diese Stellung mit der entsprechenden Speichertaste ab.

2. Möglichkeit: Direkt-Kanal-Wahl

Geben Sie direkt die Kanalnummern 35 bis 40 ein. In diesem Bereich sollte sich das Computerbild befinden. Wenn Sie es gefunden haben, speichern Sie es bitte mit der entsprechenden Speichertaste ab.

3. Möglichkeit: Fernseher mit Sendereinstellung mittels Knöpfen zum Drehen

Bei diesen Modellen befindet sich im allgemeinen ein kleiner Schalter in der Nähe des Drehknopfes, an dem die Bezeichnungen I, III und U stehen. Bitte stellen Sie diesen Schalter auf U, wenn er noch nicht in dieser Stellung steht. Nun drehen Sie den Knopf, bis Sie das Computerbild sehen.

Sollten Sie kein Bild auf Ihrem Fernseher empfangen, überprüfen Sie bitte die Verbindungen zum Computer und das Antennenkabel. Kontrollieren Sie auch, ob Ihr Computer eingeschaltet ist, dies ist an der roten Lampe Ihres Computers ersichtlich. Probieren Sie es noch einmal, nachdem Sie sich die Bedienungsanleitung Ihres Fernsehers noch einmal genau angeguckt haben. Wenn Sie auch danach keinen Erfolg haben, sollten Sie zu Ihrem Händler gehen, es könnte der Modulator des Computers defekt sein.

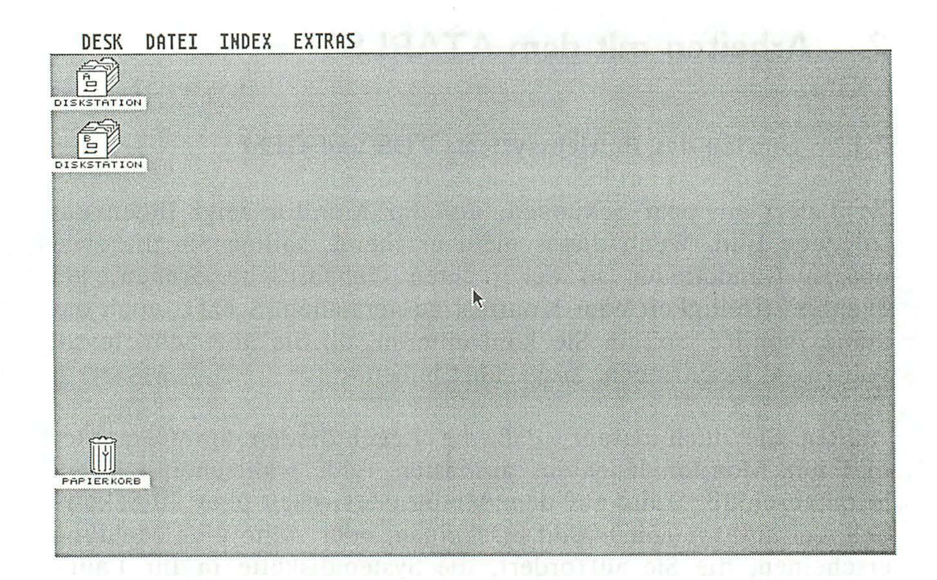
Im weiteren Verlauf des Buches sollten Sie bedenken, daß sich Ihr Fernseher wie ein Farbmonitor verhält, egal, ob Sie einen Farb- oder Schwarz-Weiß-Fernseher angeschlossen haben. Das gilt sowohl für die Bildschirmauflösungen als auch für die Farbpaletten. Bei den Farbeinstellungen, die später beschrieben werden, sollten Sie solange mit verschiedenen Farben probieren, bis Sie das für Sie optimale Bild gefunden haben.

2. Arbeiten mit dem ATARI ST

2.1 Starten des Betriebssystems TOS und GEM

Es dauert ein paar Sekunden, und Ihr Monitor zeigt Ihnen das folgende Bild. Wenn dieses nicht erscheint, sollten Sie die zwei oberen Rändelräder an der rechten Gehäuseseite drehen, um eventuell Helligkeit oder Kontrast zu verstellen. Schafft auch das keine Abhilfe, sollten Sie kontrollieren, ob Sie alle Schritte so, wie zuvor beschrieben, ausgeführt haben.

Achten Sie auch darauf, daß die Leuchtdioden am Computer und am Monitor leuchten, ansonsten fehlt wahrscheinlich ein Stromanschluß. Sollte auf dem Monitor erst nach über 30 Sekunden das untenstehende Bild erscheinen, oder sollte eine Meldung erscheinen, die Sie auffordert, die Systemdiskette in Ihr Laufwerk einzulegen, so gibt es mehrere Fehlerquellen. Überprüfen Sie zuerst, ob der Stecker des Netztrafo an das Stromnetz angeschlossen ist. Überprüfen Sie dann den Trafoanschlußstecker an dem Laufwerk. Da das Laufwerk nicht über eine Kontrolleuchte verfügt, die Ihnen anzeigt, ob das Laufwerk ein- oder ausgeschaltet ist, sollten Sie kontrollieren, ob der Einschalter auch wirklich eingeschaltet ist. Wenn Sie über das Laufwerk nach hinten links an den Schalter fassen, muß dieser nach links eingedrückt sein. Zur Kontrolle können Sie das Laufwerk auch von hinten betrachten. Neben dem Schalter ist da eine '1' für 'an' zu sehen. Eine weitere Fehlerquelle ist, daß man das Verbindungskabel zwischen Computer und Diskettenlaufwerk am Laufwerk nicht in die richtige Buchse gesteckt hat. Auf der Rückseite des Laufwerks befinden sich zwei völlig identische Buchsen. Das Kabel vom Computer muß jedoch unbedingt in die äußere Buchse, die auch mit 'in' bezeichnet ist. Sollten alle diese Weisungen nicht zum Ziel geführt haben und alle sonstigen Verbindungen vollständig und richtig angeschlossen sein, so liegt sehr wahrscheinlich ein Defekt vor.



Das Betriebssystem wurde beim Einschalten aktiviert. Was versteht man nun unter einem Betriebssystem? Kurz erklärt könnte man sagen, daß wir mit dem Betriebssystem dem Computer das einladen, was er braucht, um überhaupt arbeiten zu können. Es ist praktisch unsere Arbeitsanweisung an ihn, wie er was tun soll. Alle Meldungen, all das, was er uns mitteilt, wie wir im folgenden noch sehen werden, sind in diesem Betriebssystem enthalten. Ohne ein solches Betriebssystem kann Ihr ATARI ST gar nichts.

2.2 Der elektronische Schreibtisch: GEM DESKTOP

Sie sehen also nun eine weiße Leiste am oberen Bildschirmrand, die die Begriffe DESK, DATEI, INDEX und EXTRAS enthält.

Sie müssen sich vorstellen, diese Abbildungen entsprächen Ihrem Schreibtisch. Zwei Diskettenstationen stehen drauf, unten steht der Papierkorb. Da wir diesen Vergleich im weiteren Verlauf dieses Kapitels des öfteren gebrauchen, können wir, auf unseren Schreibtisch bezogen, die Diskettenstationen (wer hat die schon

in seinem Büro auf dem Schreibtisch stehen?) als große Karteikästen ansehen (die Bildchen auf Ihrem Monitor zeigen übrigens die Diskettenstationen ebenfalls in der Form von Karteikästen).

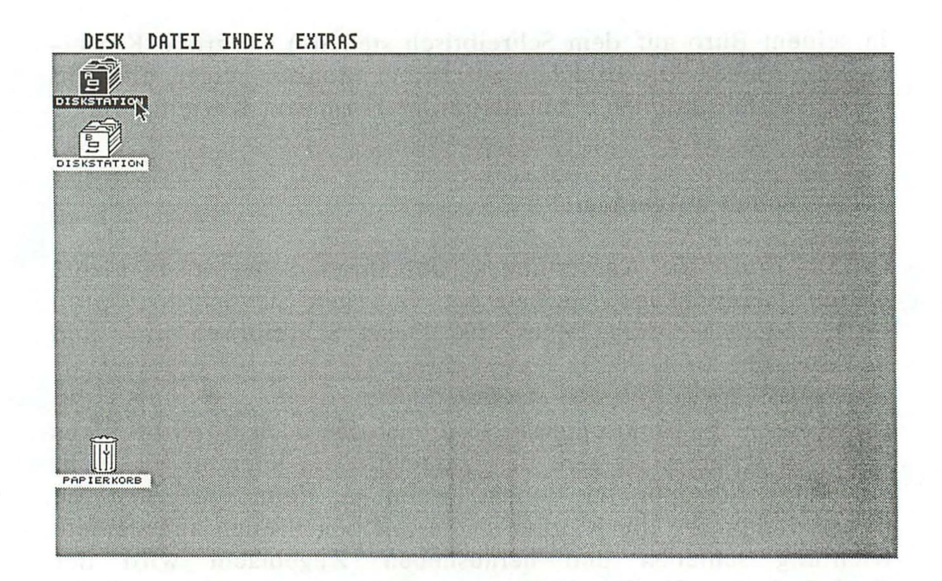
2.2.1 Schon aufgeräumt?

Gefällt Ihnen die Anordnung so auf Ihrem Schreibtisch nicht? Na gut, dann bauen wir halt um. Bewegen Sie den Mauspfeil bitte, indem Sie Ihre "Maus" auf Ihrem Schreibtisch hin- und herbewegen.

Sollte das nicht funktionieren, so drehen Sie doch bitte die Maus um, und schauen Sie sich die Unterseite an. Klemmt vielleicht ein Stück Styropor im Boden? Wenn ja, dann entfernen Sie dieses, indem Sie die Klappe in die mit den Pfeilen angedeutete Richtung schieben und herausheben. Zugemacht wird der Deckel, indem Sie ihn in die andere Richtung schieben. Sollte der Mauspfeil Ihren Bewegungen mit der Maus immer noch nicht gehorchen, überprüfen Sie bitte, ob der Anschlußstecker der Maus auch fest in Port "0" steckt. Nur wenn auch das nicht den gewünschten Erfolg bringt, so müssen Sie wohl oder übel Ihren Händler aufsuchen, da wahrscheinlich etwas defekt ist.

Lassen Sie den Mauspfeil ruhig noch etwas kreisen. Beschränken Sie sich dabei aber bitte auf den grauen Hintergrund, und vermeiden Sie den Kontakt mit der oberen weißen Zeile. Sollte Sie dennoch einmal in die obere Leiste kommen, so werden Sie plötzlich eine Art "heruntergelassenes Rollo" erkennen. Diese Rollos werden wir später besprechen, doch eins vorweg, ist ein Rollo erst einmal auf, so können Sie es schließen, indem Sie Ihren Mauspfeil auf ein freies Plätzchen auf Ihrem Bildschirm bewegen und die linke Maustaste betätigen.

Bewegen Sie den Mauspfeil jetzt auf das Symbol einer Diskettenstation und drücken nun kurz den linken Mausknopf. Der ST zeigt an: 'Ich hab's verstanden', denn nun ist die Diskettenstation bzw. der Karteikasten anders dargestellt als vorhin: alles, was vorher weiß war, ist nun schwarz, und umgekehrt.



Nun drücken Sie noch einmal die Maustaste, halten sie aber gedrückt und bewegen nun die Maus auf Ihrem Bildschirm ... pardon, auf Ihrem Schreibtisch, nach rechts. Mit dem wohlbekannten Pfeil ziehen Sie nun mit den Umrissen der Diskettenstation auf dem Bildschirm umher. Vermeiden Sie den Kontakt mit anderen Symbolen und der oberen Leiste, da Sie damit ungewollt Aktionen starten, die wir erst im weiteren Verlauf des Buches besprechen wollen.

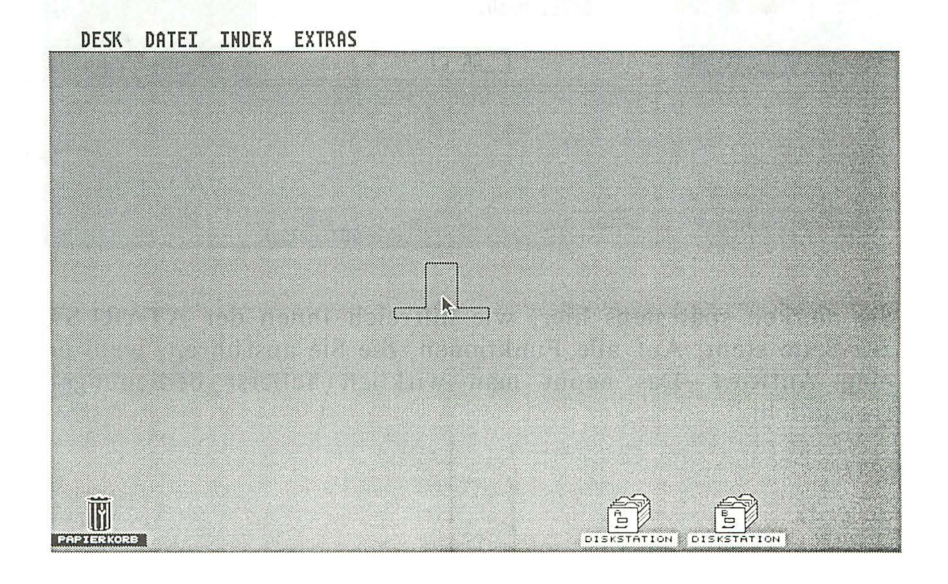
Lassen Sie schließlich die Maustaste los, so wird genau dort, wo der Umriß zuletzt erschienen ist, die vollständige Darstellung der Diskettenstation angezeigt. Sie haben also durch die Bewegung der Maus den Umbau der Utensilien auf Ihrem Schreibtisch vorgenommen, indem Sie erst nur den Umriß fortbewegt haben, danach aber Ihr ST den Umriß in die entsprechenden Formen an der neuen Position übertragen hat, und die Darstellung an der ursprünglichen Bildschirmposition verschwunden ist. Sollten Sie dennoch einmal zwei Symbole aufeinander legen, so muß man zwei Fälle unterscheiden:

a) Sie haben die beiden Diskstationen aufeinander gelegt. Es erscheint ein Bild in der Bildschirmmitte, in dem Sie in der unteren Hälfte zwei Kästchen, eins mit "OK" und eines mit "Abbruch" sehen. Bewegen Sie Ihren Pfeil auf das Kästchen mit Abbruch, und drücken Sie die linke Maustaste. Das Bild verschwindet wieder, und Sie können normal weiterarbeiten.

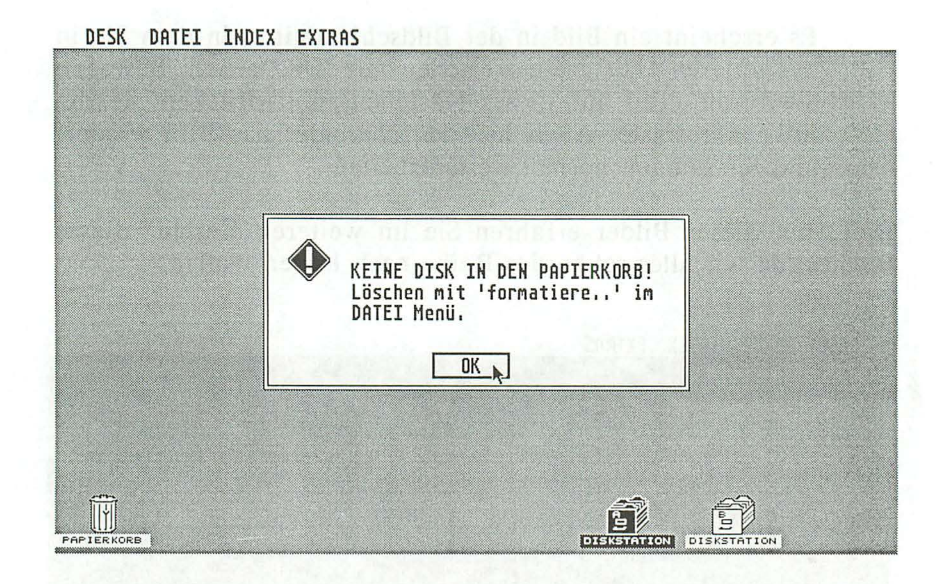
b) Sie haben eine Diskstation auf den Papierkorb gelegt oder umgekehrt:

Es erscheint ein Bild in der Bildschirmmitte, in dem Sie in der unteren Hälfte ein Kästchen mit "OK" sehen. Bewegen Sie Ihren Pfeil auf dieses Kästchen, und drücken Sie die linke Maustaste. Auch hier verschwindet das Bild wieder, und Sie können normal weiterarbeiten.

Den Sinn dieser Bilder erfahren Sie im weiteren Verlauf dieses Buches, da wir alles schön der Reihe nach lernen wollen.



Bewegen Sie ruhig auch den Papierkorb an eine andere Bildschirmstelle. Ein herrliches Spielzeug, dieses Betriebssystem, nicht wahr? Bisher haben wir mit diesem Verschieben der Bilder (wir sprechen hier auch von "Icons") auf dem Monitorbildschirm erst einen kleinen Teil dieses Betriebssystems kennengelernt. Übrigens: In Wirklichkeit läuft diese 'Verschieberei' eigentlich nicht auf dem Monitor, sondern unterhalb der Tastatur im Inneren Ihres ATARI ST ab, der Monitorbildschirm dient für diese Funktion lediglich als Anzeige.



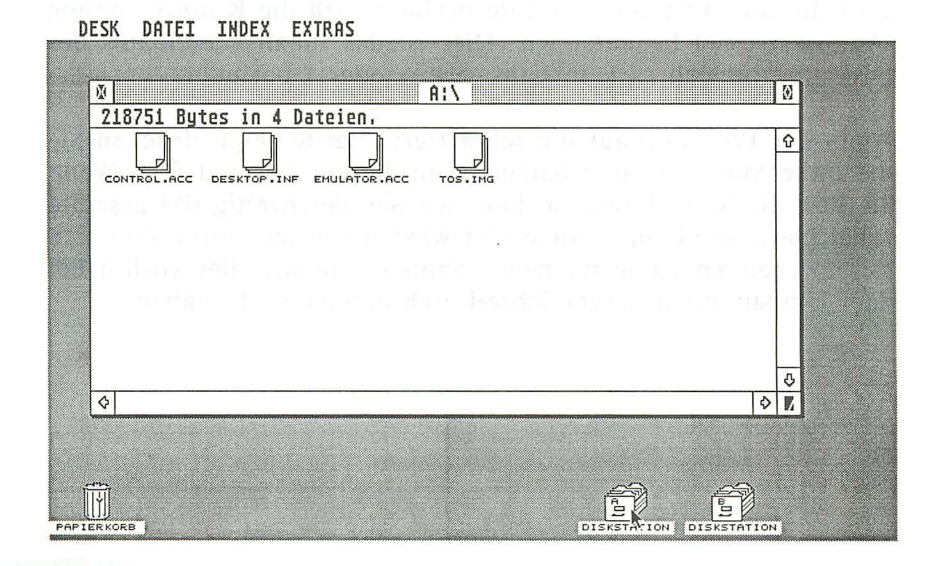
Sie merken spätestens hier, wie hilfreich Ihnen der ATARI ST zur Seite steht. Auf alle Funktionen, die Sie ausführen, weiß er eine Antwort. Das nennt man wirklich äußerst bedienungsfreundlich!

2.2.2 Das Inhaltsverzeichnis der Diskette

Nun wollen wir uns doch einmal ansehen, was denn in unserer Diskettenstation bzw. in unserem Karteikasten enthalten ist. Steuern Sie zu diesem Zweck mit Ihrem Pfeil auf das Bild von Diskettenstation A zu und drücken anschließend einmal die linke Maustaste.

Haben Sie dies richtig gemacht, färbt sich die Diskettenstation erneut schwarz, d.h. der ST hat verstanden, daß Sie etwas mit dieser Diskettenstation tun wollen. Drücken Sie nun noch zweimal ganz kurz nacheinander die linke Maustaste, und aus dem Symbol der Diskettenstation A schießt blitzschnell ein Blatt Papier, das sich vor unseren Augen entfaltet.

Sollte nichts passieren, so drücken Sie noch erneut zweimal schnell hintereinander die linke Maustaste. Dieses Verfahren nennt man "Doppelklick". Am Anfang haben Sie vielleicht etwas Schwierigkeiten, so schnell hintereinander die linke Maustaste zu betätigen, bald aber wird dieser Doppelklick auf Anhieb funktionieren.



So als wenn Sie in Ihren Karteikasten einzelne Deckblätter eingelegt hätten, die Ihnen die Orientierung im Karteikastensystem erleichtern, enthält die Diskette, die sich zur Zeit in der Diskettenstation befindet, auch mehrere Teile, einzelne Programme oder Datensätze, die Sie nun auf dem Bildschirm in Ruhe betrachten können, um sich anschließend für eine sinnvolle Auswahl zu entscheiden.

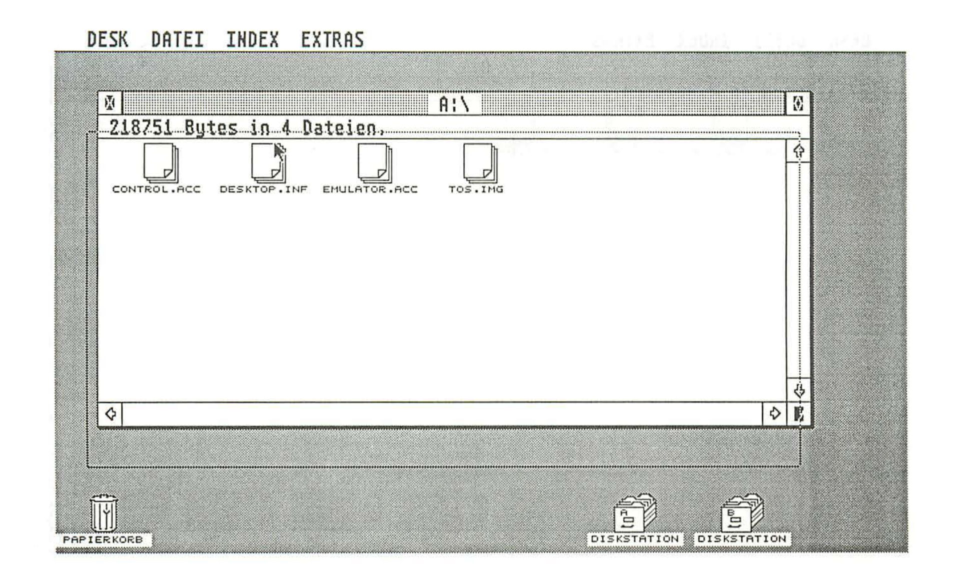
Zusammengefaßt könnte man diese Auflistung auch als Inhaltsverzeichnis der Diskette bezeichnen, die im Moment in der Diskettenstation A liegt.

Schauen wir uns das Papier etwas genauer an. Sie erinnern sich noch an vorhin, als wir unseren Schreibtisch umgeräumt haben? Das gleiche können wir nun auch mit den Programmen und Datensätzen auf unserem Schreibtisch/Bildschirm tun.

2.2.2.1 Verschieben

Bewegen Sie den Pfeil mit Hilfe der Maus hierzu doch einfach mal auf die schraffierte Leiste des angezeigten Papiers. Übrigens: In der Mitte dieser Kante befindet sich die Kennzeichnung 'A:', was soviel bedeutet wie: Dies ist das Inhaltsverzeichnis der Diskette, die sich z.Zt. in Laufwerk A oder 1 befindet.

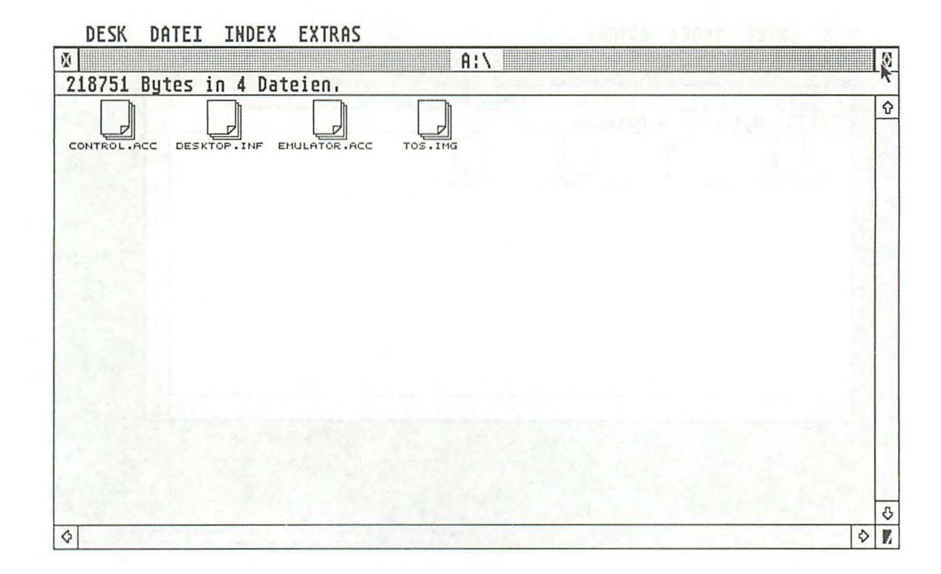
Wenn der Pfeil also auf die schraffierte Leiste zeigt, drücken Sie die linke Maustaste und halten sie in diesem Zustand fest. Wenn Sie nun die Maus bewegen, bewegen Sie gleichzeitig das gesamte Inhaltsverzeichnis mit. Angezeigt wird Ihnen das durch den sich mitbewegenden Papierrahmen - ähnlich wie wir dies vorhin bei dem Umbau auf unserem Schreibtisch bereits erlebt haben.



Lassen Sie den linken Mausknopf los, sieht es aus, als wenn das Bild dem Rahmen nachkommen würde, und in Sekundenschnelle ist unser Blatt Papier verschoben. Vielleicht verdeckt es nun sogar die Bilddarstellungen der Diskettenstationen oder des Papierkorbs.

2.2.2.2 Ausfüllen

Was bedeuten die anderen Kennzeichnungen am Rand unseres Inhaltsverzeichnisses? Bewegen Sie hierzu den Pfeil mit der Maus zuerst in die rechte, obere Ecke des Papiers und drücken den linken Mausknopf (bitte diesmal nicht festhalten, sondern nur einmal anklicken). Nun vergrößert sich die Darstellung des weißen Papiers, bis es den gesamten Bildschirm ausfüllt.



Noch einmal dieses kleine Karo angeklickt, und der Vorgang von gerade eben wird wieder rückgängig gemacht ... wir sehen wieder nur einen Teil unseres Inhaltsverzeichnisblattes.

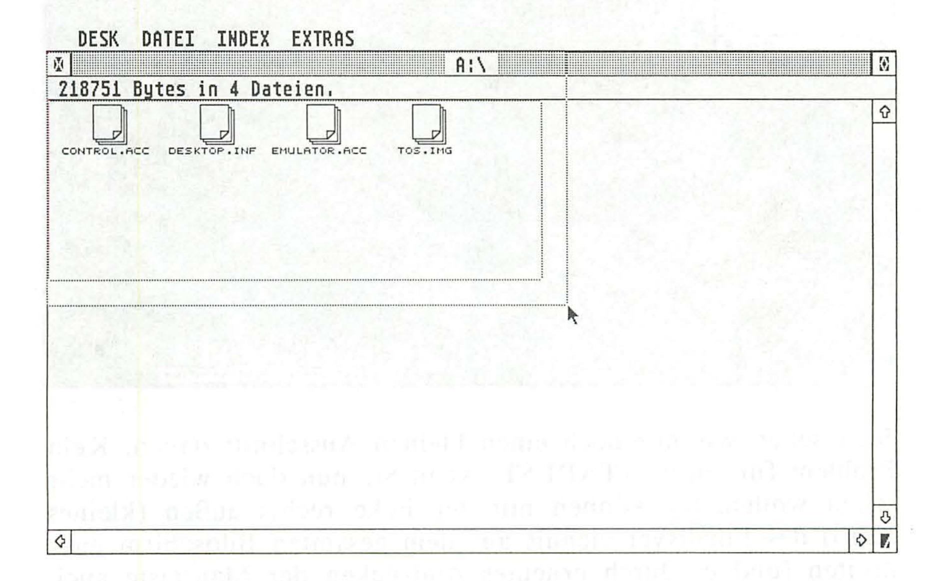
Also:

Das kleine Karo in der rechten oberen Ecke ist ein An-/Ausschalter. Hiermit vergrößern wir beim ersten Anklicken das Papier, so daß es den gesamten Bildschirm ausfüllt, beim erneuten Anklicken kehrt die Darstellung wieder in den Urzustand zurück.

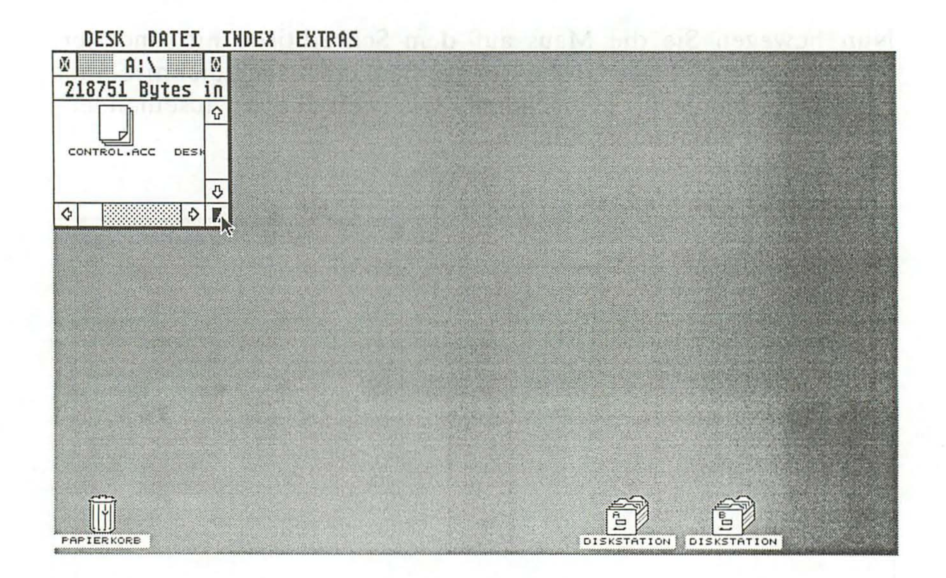
2.2.2.3 Vergrößern und Verkleinern

Mit dem kleinen Symbol rechts unten können wir nun das Inhaltsverzeichnis auseinanderziehen oder auch zusammendrücken, als wäre es aus Gummi. Bewegen Sie den Mauspfeil auf die Ecke rechts unten von unserem Inhaltsverzeichnisblatt, drücken anschließend die linke Maustaste und halten sie auch weiterhin gedrückt.

Nun bewegen Sie die Maus auf dem Schreibtisch hin und her. Sie können somit die Größe des weißen Inhaltsverzeichnisblattes fließend verändern, je nachdem, wie weit Sie es auseinanderziehen bzw. zusammenschieben.



Damit wir nun auch noch die letzten Kennzeichnungen am Rand unseres weißen Blattes kennenlernen, stauchen Sie doch einfach das Papier mit Hilfe der Gummiecke soweit zusammen, daß es nicht mehr kleiner geht. Lassen Sie anschließend den Mausknopf los und ... es ist wirklich winzig geworden, unser Inhaltsverzeichnis.



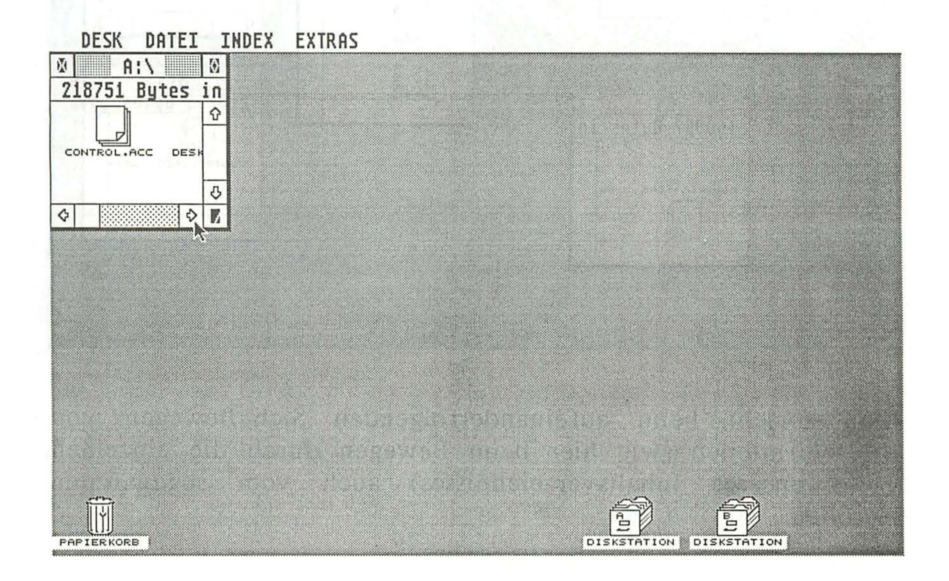
Jetzt sehen wir nur noch einen kleinen Ausschnitt davon. Kein Problem für Ihren ATARI ST, wenn Sie nun doch wieder mehr sehen wollen: Sie können mit der Ecke rechts außen (kleines Karo) das Inhaltsverzeichnis auf dem gesamten Bildschirm ausbreiten (und es durch erneutes Andrücken der Maustaste auch wieder auf seine ursprüngliche Größe verkleinern) oder es mit der Gummiecke rechts unten auseinanderziehen, d.h. es ausbreiten.

2.2.2.4 Rollen mit den Pfeilen

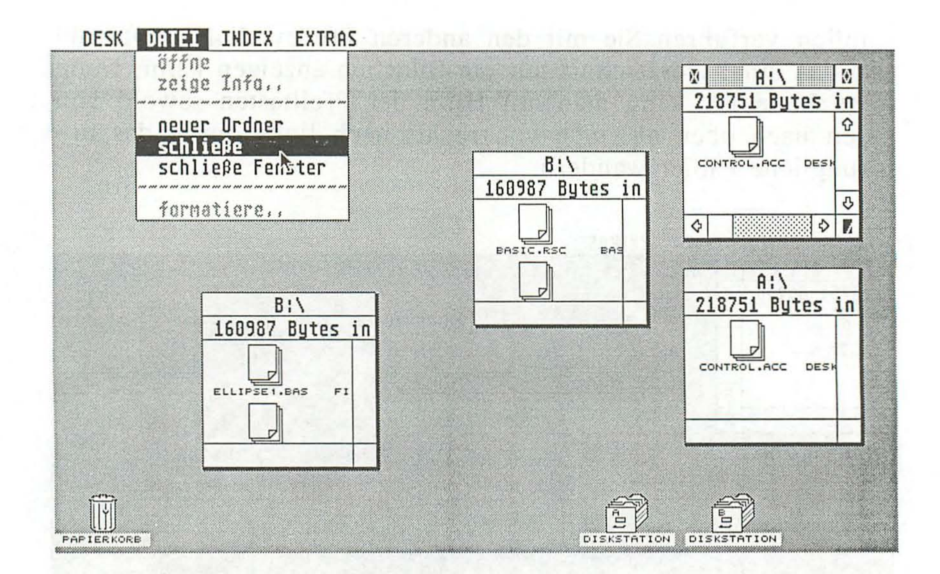
Das alles ist nicht nötig, wenn Sie sich mit Hilfe der vier Pfeilsymbole (am Rand unseres Inhaltsverzeichnisblattes) Darstellung für Darstellung auf dem verkleinerten Bildschirm aufeinanderfolgend anzeigen lassen.

Sie haben jetzt also das Inhaltsverzeichnis so klein zusammengedrückt, wie es geht. Bewegen Sie nun den Mauspfeil z.B. auf das Pfeilsymbol rechts unten (links von unserer Gummiecke) und drücken Sie anschließend kurz den Mausknopf: Schwups ... das nächste Bild unseres Inhaltsverzeichnisses wird sichtbar gemacht.

Ähnlich verfahren Sie mit den anderen Pfeilsymbolen. Obwohl also das Inhaltsverzeichnis nur ein Bildchen anzeigen kann, wenn es so klein ist, kann man mit Hilfe der Pfeiltasten sowohl von unten nach oben als auch von rechts nach links durch das ursprüngliche Papier wandern.



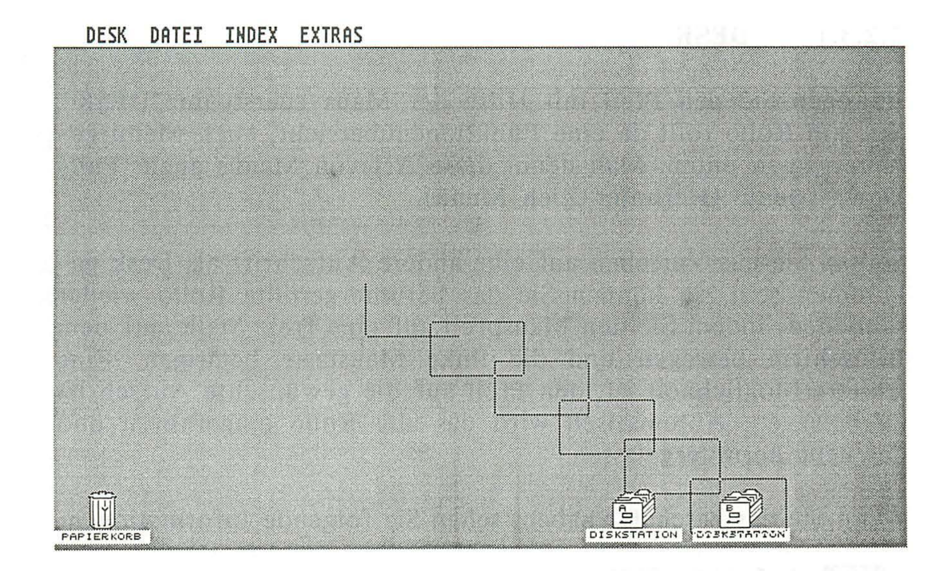
Vielleicht fragen Sie sich, wozu das gut sein soll? Stellen Sie sich einmal vor, Sie haben ganz viele Inhaltsverzeichnisse auf einmal auf Ihrem Bildschirm. Damit wirklich für alle genug Platz ist, verkleinert man dann jedes einfach, damit es kein Gedrängel gibt.



Man spricht beim aufeinanderfolgenden Sich-Bewegen von Bildschirmteilen (wie hier beim Bewegen durch die einzelnen Teile unseres Inhaltsverzeichnisses) auch vom sogenannten 'Scrollen'.

2.2.2.5 Jetzt aber Schluß!

Die letzte Ecke unseres Inhaltsverzeichnisses, links oben, in der man ein X sieht, ist für das völlige Verschwinden des Inhaltsverzeichnisblattes vom Bildschirm verantwortlich. Diesen Vorgang nennt man auch Schließen des Fensters. Drücken Sie hier, nachdem Sie den Mauspfeil auf diese Ecke gerichtet haben, die linke Taste Ihrer Maus: Das weiße Papier mit dem Inhaltsverzeichnis verschwindet, und unser Schreibtisch sieht wieder ganz sauber aus. Wie ganz am Anfang sind auf dem Schreibtisch nun nur noch zwei Karteikästen/Diskettenstationen und ein Papierkorb zu sehen. Das Inhaltsverzeichnisblatt ist in Diskettenstation A verschwunden.



Also:

Die Schließecke läßt das Inhaltsverzeichnis wieder in die Diskettenstation, aus der es gekommen ist, verschwinden, löscht also den Bildschirm, und nur noch das Ausgangsbild ist zu sehen.

2.2.3 Die Pull-Down-Menüs

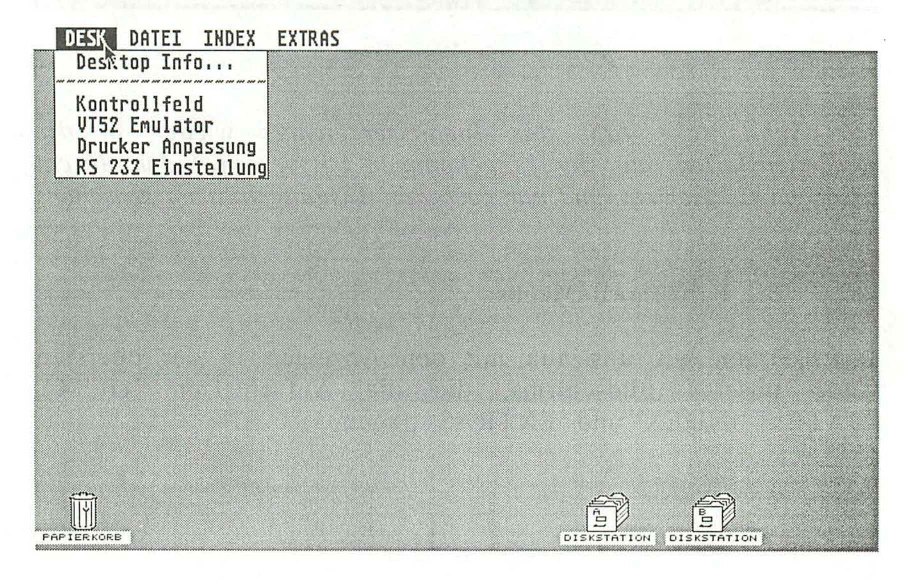
Beschäftigen wir uns nun mit den Anzeigen in der obersten Zeile unseres Bildschirms, die die Aufschriften 'DESK', 'DATEI', 'INDEX' und 'EXTRAS' tragen.

2.2.3.1 DESK

Bewegen Sie den Pfeil mit Hilfe der Maus zuerst auf 'DESK'. Wie ein Rollo rollt da eine Funktionenübersicht, auch Menü genannt, nach unten. Man nennt diese Art von Menüs auch "Pull-Down-Menü" (Herunter-Zieh-Menü).

Sollten Sie aus Versehen auf eine andere Aufschrift als Desk gekommen sein, so können Sie das heruntergerollte Rollo wieder schließen, indem Sie den Mauspfeil auf eine freie Stelle auf dem Bildschirm bewegen und die linke Maustaste betätigen. Eine weitere Möglichkeit ist, den Pfeil auf die gewünschte Aufschrift zu bewegen. Automatisch wird das alte Rollo eingefahren und das neue heruntergelassen.

Wenn Sie das geschafft haben, sehen Sie folgende Informationen:

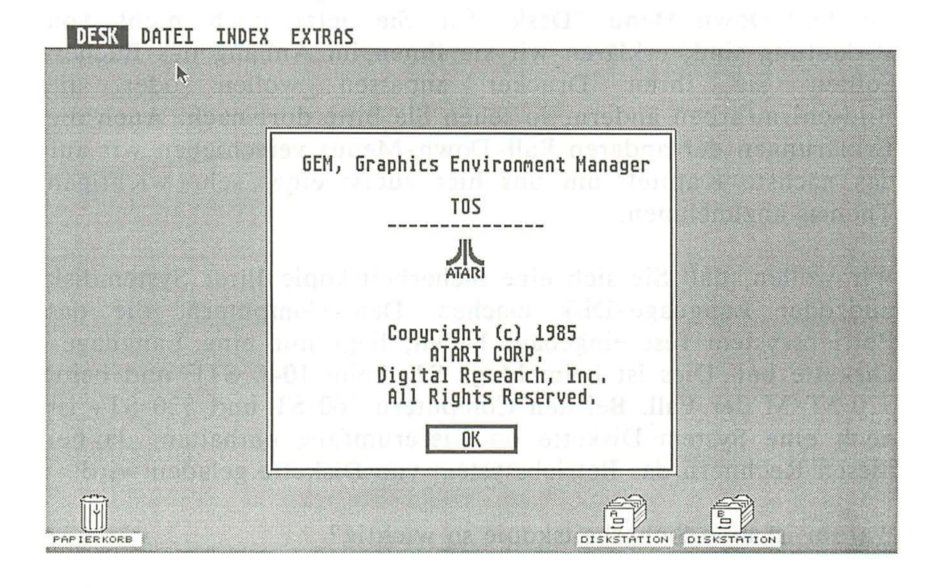


Es ist möglich, daß sowohl die Reihenfolge als auch die Anzahl der angezeigten Menüpunkte Ihres Bildschirms von unserer Abbildung abweicht. Dies ist mit unterschiedlichen Betriebssystemversionen zu erklären, ändert aber an der grundsätzlichen Funktionsweise nichts.

Gehen wir mit dem Pfeil auf die einzelnen Wörter im Pull-Down-Menü, merkt das unser ST sofort: die Wörter werden aktiviert, d.h. wie unsere vorhin mit Mausklick angewählte Diskettenstation werden sie schwarz dargestellt. Schauen wir uns an, was sich hinter dem ersten Begriff verbirgt:

Desktop Info

Zuerst richten Sie den Mauspfeil also auf 'Desktop Info ...' und drücken anschließend die linke Maustaste. Augenblicklich erscheint in der Bildschirmmitte folgende Anzeige:



Was Sie nun sehen können, ist eine Information über das Betriebssystem TOS und die Benutzerschnittstelle GEM, die Sie vorhin in Ihren ST von Diskette eingeladen hatten bzw. die in Ihrem Rechner bereits eingebaut ist. Sollte sich das Bild in der Copyright-Meldung mit anderen Jahreszahlen zeigen, so ist in Ihrem Rechner das sogenannte Blitter-TOS eingebaut. Dies ist beim Mega ST und 1040 STF neuerer Bauart der Fall. Auf die geringen Unterschiede weisen wir bei den jeweiligen Kapitelteilen hin.

Sicher haben Sie von GEM schon einiges gehört. GEM (Graphics Environment Manager; Manager für die grafische Umgebung) ist genau der Teil des vorher von Ihnen geladenen Betriebssystems, der die grafische Steuerung verschiedenster grafischer Funktionen erst möglich macht (z.B. die Funktion der vier Ecken der Inhaltsverzeichnisdarstellung).

Haben Sie die Information am Bildschirm eingehend studiert, bewegen Sie den Mauspfeil auf die OK-Anzeige und drücken ein weiteres Mal die linke Maustaste. Sofort verschwindet dieses Fenster wieder vom Bildschirm.

Da die weiteren Punkte 'Druckeranpassung' und 'Kontrollfeld' im Pull-Down-Menü "Desk" für Sie jetzt noch nicht von Bedeutung sind, erklären wir sie Ihnen im Anhang des Buches. Sollten Sie Ihren Drucker anpassen wollen oder die Bildschirmfarben ändern, so sehen Sie bitte dort nach. Auch die Erklärungen der anderen Pull-Down-Menüs verschieben wir auf das nächste Kapitel, um uns hier zuerst eines sehr wichtigen Themas anzunehmen.

Wir wollen, daß Sie sich eine Sicherheitskopie Ihrer Systemdisk und/oder Language-Disk machen. Den Computern, die das Betriebssystem fest eingebaut haben, liegt nur eine Language-Diskette bei. Dies ist beim Mega ST, beim 1040 STF und beim 520 ST/M der Fall. Bei den Computern 260 ST und 520 ST+ ist noch eine System-Diskette im Lieferumfang enthalten, da bei diesen Rechnern das Betriebssystem von Diskette geladen wird.

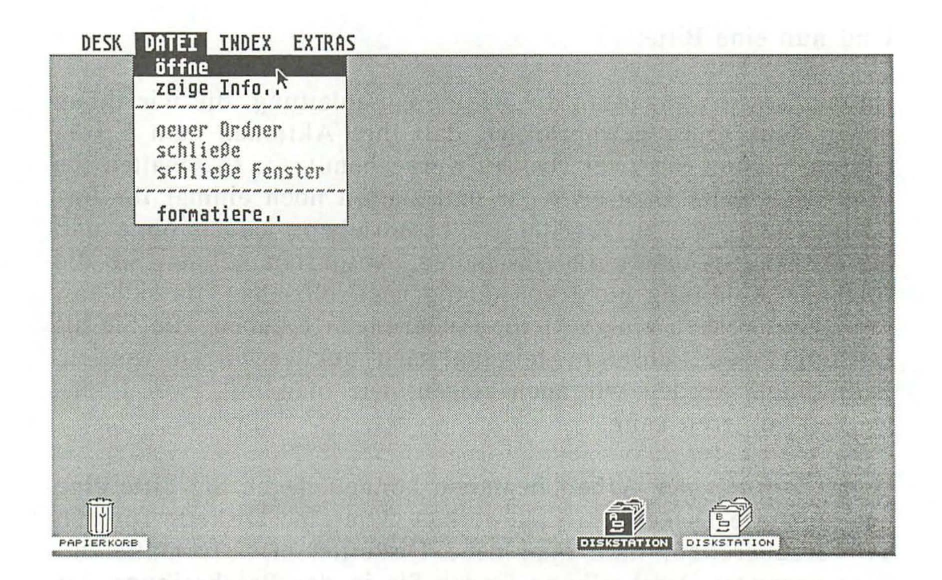
Warum ist eine Sicherheitskopie so wichtig?

Nun, stellen Sie sich vor, mit Ihrer Originaldiskette passiert irgend etwas. Jemand verschüttet aus Versehen Kaffee darauf oder aber die Diskette war einem magnetischen Feld ausgesetzt (z.B. zu nahe am Monitor oder an den Netzgeräten gelagert). Sie könnten mit Ihrem Computer nicht mehr arbeiten. Haben Sie jedoch eine Sicherheitskopie, so können Sie die Originaldiskette im Schrank verstauen und ohne Sorge mit der Sicherheitskopie arbeiten.

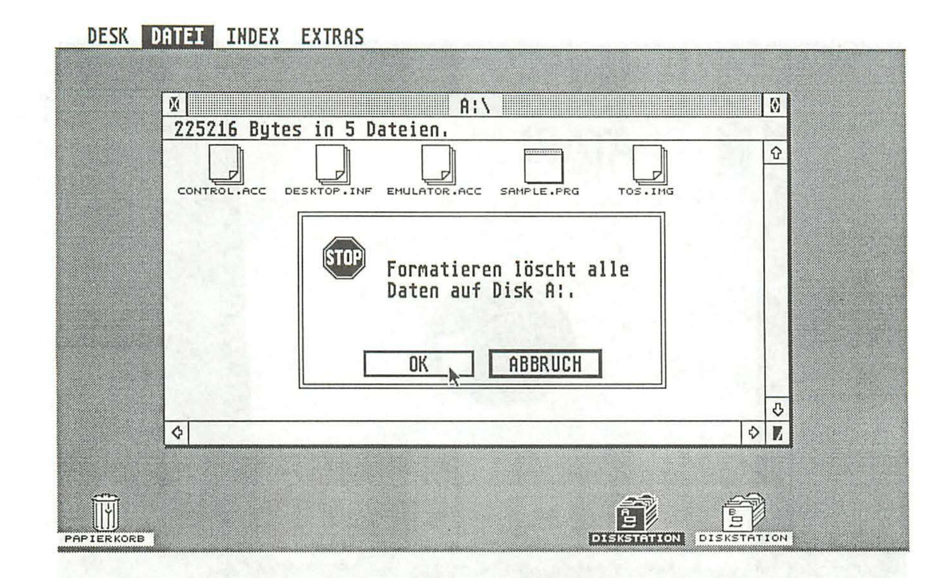
Und nun eine Bitte:

Halten Sie sich genau an die Bedienungsanleitung, die wir Ihnen geben. Nur so ist gewährleistet, daß Ihre Aktionen zum Erfolg führen. Sollten Sie zwei Disklaufwerke benutzen, so schalten Sie bitte das zweite Laufwerk aus und starten noch einmal Ihr Betriebssystem, wie in Kapitel 1.5 beschrieben, jedoch ohne daß Sie das zweite Laufwerk einschalten. Ansonsten können Sie die folgende Anleitung nicht vollständig nachvollziehen, da sich mit zwei Laufwerken einige kleine Änderungen ergeben, die Sie als Einsteiger jetzt unserer Meinung nach nur verwirren werden. Bald schon werden wir auch zeigen, wie man mit zwei Laufwerken kopieren kann.

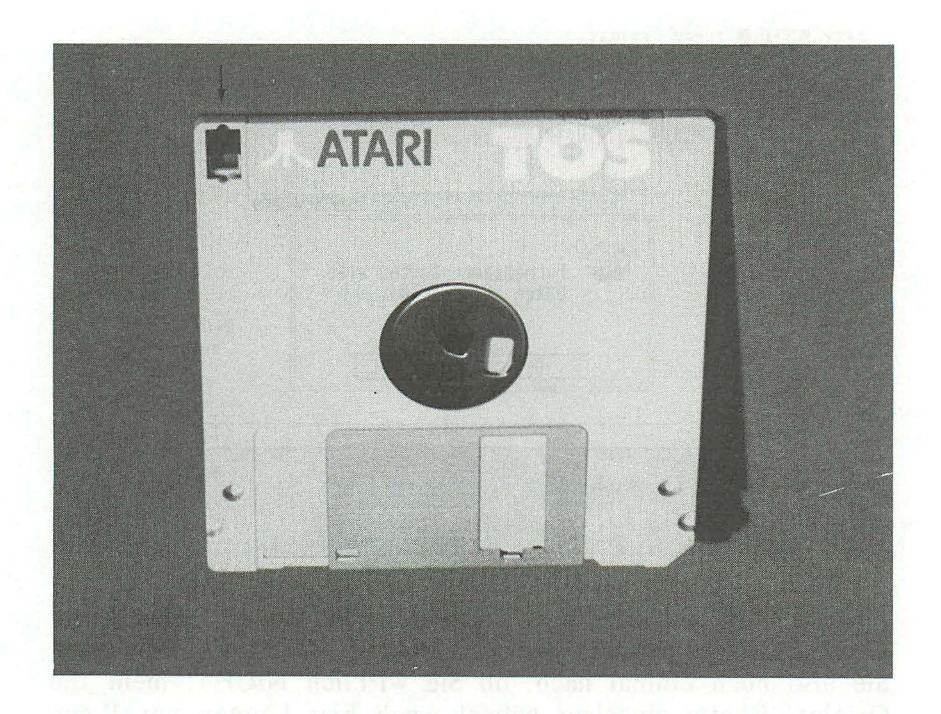
Damit wir mit der Arbeit beginnen können, legen Sie bitte eine neu gekaufte Diskette in Laufwerk A. Um eine Diskette für den ST vorzubereiten, muß man neue Disketten erst "formatieren". Eine genauere Beschreibung finden Sie in der Beschreibung der weiteren Pull-Down-Menüs. Zum Formatieren bewegen Sie den Mauspfeil auf Diskstation A und aktivieren diese durch Betätigen der linken Maustaste. Nun bewegen Sie den Mauspfeil auf das Pull-Down-Menü "Datei".



Sie haben sicher den Menüpunkt "formatiere.." schon entdeckt. Sie sehen aber auch, daß einige Menüpunkte nicht schwarz, sondern grau erscheinen. Diese Menüpunkte lassen sich auch nicht aktivieren, wenn Sie mit Ihrem Mauspfeil darauf gehen. Der ATARI ST zeigt Ihnen mit dieser Darstellung an, daß diese Punkte zur Zeit nicht zur Verfügung stehen. So ist gewährleistet, daß Sie keine Aktionen durchführen können, die zur Zeit nicht erlaubt sind. Hätten wir beispielsweise nicht vorher Diskstation A aktiviert, so könnten wir "formatieren" gar nicht anwählen, da der Computer ja nicht wissen kann, auf welchem Laufwerk die Aktion ausgeführt werden soll. Aktivieren wir aber nun in bekannter Weise den Menüpunkt "formatieren..". Wie Sie sicher noch wissen, müssen wir dazu den Mauspfeil auf das Wort "formatieren.." bewegen, bis sich dieses schwärzt, und anschließend die linke Maustaste betätigen. Wenn das alles geklappt hat, sollte sich folgendes Bild ergeben:



An der Warnbox erkennen Sie, welche Folgen es haben kann, wenn Sie ungewollt eine wichtige Diskette formatieren. Schauen Sie also noch einmal nach, ob Sie wirklich NICHT! mehr die Originaldiskette eingelegt haben! Auch hier können wir Ihnen wieder einen guten Typ geben. Disketten kann man vor dem Überschreiben schützen. Dafür wird vom Hersteller eine Art kleiner Schalter in die Diskettenhülle eingebaut, den Sie erkennen können, wenn Sie die Diskette von der Rückseite her betrachten.



Die Funktion dieses kleinen Stellplättchens ist nun die, daß sich in einer Stellung ein kleines Loch öffnet und in der anderen Stellung schließt. Diese Öffnung wird nun bei jedem Schreib- und Formatiervorgang kontrolliert. Können Sie durch die Öffnung durchgucken, so kann nicht auf die Diskette geschrieben werden, sie ist damit schreibgeschützt. Bitte stellen Sie bei allen Originaldisketten das Schreibschutzplättchen so, daß Sie durch die Öffnung schauen können. Das ist der beste Schutz dagegen, daß Sie sich auf einmal eine wichtige Diskette überschreiben.

Nun, da wir sicher sind, daß wir formatieren wollen, bestätigen wir die Warnbox mit "OK". Sofort erscheint eine weitere Box, diesmal jedoch eine, in der wir etwas eingeben können.

FORMATIEREN

FORMATIEREN

AUSGANG

Stationskennung: A: FORMAT

Disketten Name: ________.___

Format:

Einseitig Zweiseitig

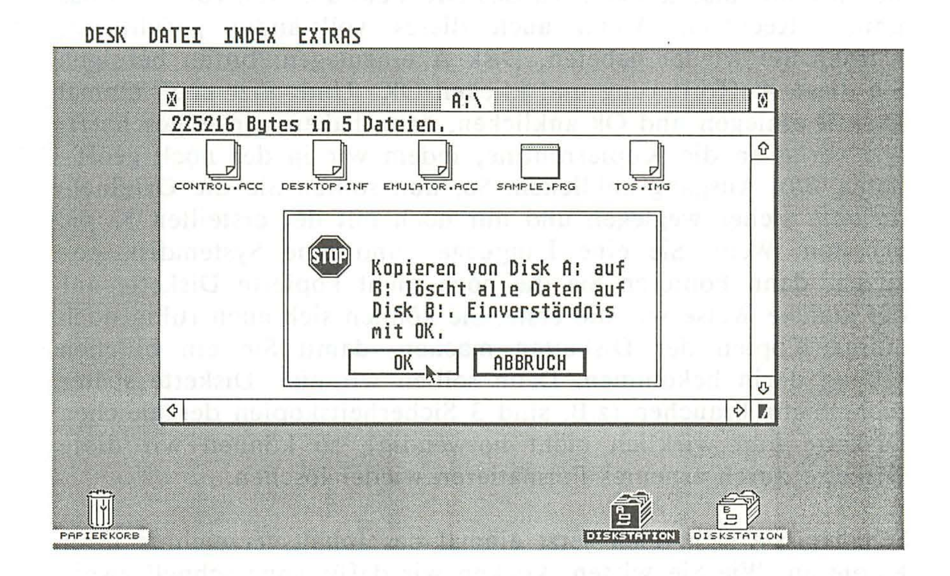
Bitte tragen Sie jetzt noch keinen Diskettennamen ein, und lassen Sie das Format auf "Einseitig" stehen. Haben Sie aus Versehen "Zweiseitig" betätigt, so können Sie durch Aktivieren von "Einseitig" diese Aktion rückgängig machen. Die Formatierroutine starten Sie jetzt, indem Sie den Mauspfeil auf das Kästchen "FORMAT" bewegen und einmal mit der linken Maustaste klicken. Nun beginnt das Laufwerk zu arbeiten. Auf dem Bildschirm erscheint ein neues Bild, an dem Sie den Stand der Formatierung ablesen können. Je mehr sich das Rechteck füllt, desto weiter ist die Formatierung fortgeschritten, wenn es voll ist, ist die Diskette fertig formatiert. Dann erscheint eine andere Box, die Ihnen mitteilt, wieviel Speicherplatz auf der Diskette frei ist. Bitte kontrollieren Sie, ob die Angabe der freien "Bytes" 357376 Bytes beträgt. Sollte das nicht der Fall sein, müssen Sie die Diskette noch einmal formatieren, sollte sie dann immer noch nicht diesen Wert anzeigen, ist wahrscheinlich die Diskette nicht ganz in Ordnung, wiederholen Sie den Punkt dann bitte mit einer anderen Diskette. Die Box verlassen Sie, indem Sie auf das Kästchen "OK" klicken. Sie sehen, die Bedienung mit den

Boxen gehorcht immer den gleichen Gesetzen. Ist alles klar mit der Diskette, verlassen wir die Formatierbox, indem wir auf das Kästchen "Ausgang" klicken.

Nun haben wir wieder den ganz normalen Bildschirminhalt auf dem Bildschirm, alle Boxen sind verschwunden. Bis jetzt haben wir die neue Diskette nur für den ST vorbereitet, jetzt müssen wir die Language-Disk (System-Disk) noch darauf kopieren. Legen Sie dazu die Language-Diskette (System-Disk) in das Diskettenlaufwerk, und denken Sie an den Schreibschutz. Jetzt aktivieren wir Diskstation A, drücken die linke Maustaste und halten sie fest. Genauso wie wir unseren Schreibtisch umgebaut haben, können wir jetzt die Umrisse der Diskstation A auf die Diskstation bewegen und dort die linke Maustaste loslassen. Achten Sie bitte darauf, daß Sie die Maustaste erst loslassen, wenn sich auch Diskstation B schwärzt und damit signalisiert, daß die gewünschte Aktion erkannt wurde. Sie werden sicher schnell feststellen, daß dazu die Mauspfeilspitze genau auf das Symbol der Diskstation B zeigen muß.

Ein wichtiger Hinweis: Wenn Sie ein Symbol ganz außen fassen und es überlappend auf ein anderes Symbol legen, ohne daß die Mauspfeilspitze das zweite Symbol berührt und sich dieses nicht schwärzt, kann es sein, daß beim Loslassen der linken Maustaste ein Symbol scheinbar vom Bildschirm verschwunden ist. In Wirklichkeit liegen die Symbole jetzt übereinander, und Sie können nur das obere Symbol sehen. Nehmen Sie in diesem Fall das sichtbare Symbol, und bewegen Sie es auf eine freie Stelle auf dem Bildschirm. Nun ist das "verschwundene" Symbol wieder zu sehen.

Wenn Sie unsere Anweisungen genau befolgt haben, sollen auf Ihrem Bildschirm folgende Box erscheinen:



Auch hier müssen wir bestätigen, daß wir kopieren wollen. Sie sehen, auch Ihr ST ist daran interessiert, daß Ihnen kein Fehler unterläuft. Klicken wir also das OK-Feld an, und sofort erscheint eine neue Box, die uns anzeigt, von welcher Diskstation wir auf welche kopieren. In unserem Fall sollte von Diskstation A auf Diskstation B kopiert werden. Klicken Sie jetzt bitte das Feld Kopieren an. Nun sehen Sie eine Informationsbox ähnlich der beim Formatieren. Auch hier können Sie anhand der Rechtecke sehen, wie weit Sie mit der Kopie sind. Das obere Rechteck zeigt Ihnen an, wieviel schon gelesen wurde, das untere, wieviel auf die neue Diskette geschrieben wurde. Nachdem das obere Rechteck vollständig gefüllt ist, erscheint eine Box, die Sie bittet, Disk B in Floppy A einzulegen. Das ist damit zu erklären, daß wir ja nur ein Floppylaufwerk angeschlossen haben (oder ein zweites angeschlossenes Laufwerk bis jetzt noch ausgeschaltet gelassen haben). Der Computer bezeichnet also Ihre Originaldiskette mit A und die Sicherheitskopie mit B. Befolgen

Sie die Anweisung Ihres Computers und legen die neue Diskette ein, die Sie vorhin formatiert haben. Jetzt fehlt nur noch die Bestätigung, also klicken wir das OK-Feld an. Nun füllt sich das untere Rechteck. Wenn auch dieses vollständig gefüllt ist, werden Sie wieder gebeten, Disk A einzulegen, bitten befolgen Sie diese Aufforderung und klicken OK. Und jetzt noch einmal Disk B einlegen und Ok anklicken, dann haben wir es geschafft, wir verlassen die Kopierroutine, indem wir in der noch geöffneten Box Ausgang anklicken. So, nun sollten Sie die Originaldiskette sicher weglegen und nur noch mit der erstellten Kopie arbeiten. Wenn Sie eine Language- und eine Systemdisk besitzen, dann kopieren Sie die noch nicht kopierte Diskette auf die gleiche Weise wie die erste. Sie können sich auch ruhig noch einige Kopien der Disketten machen, damit Sie ein bißchen Übung darin bekommen. Denn sollten wir eine Diskette später nicht mehr brauchen (z.B. sind 3 Sicherheitskopien der gleichen Diskette nun wirklich nicht notwendig), so können wir diese Diskette durch erneutes Formatieren wieder löschen.

Schauen Sie sich doch jetzt einmal das Inhaltsverzeichnis Ihrer Kopie an. Wie Sie wissen, klicken wir dafür ganz schnell zweimal hintereinander mit der linken Maustaste auf Diskstation A. Jetzt wundern Sie sich sicher, daß Sie erneut aufgefordert werden, Diskette A in Diskstation A einzulegen, doch erinnern Sie sich daran, daß wir zuletzt die Kopie, die der Computer mit B bezeichnet hatte, in Laufwerk A gelegt hatten. Klicken Sie also nur das OK-Feld an, und es erscheint das Inhaltsverzeichnis Ihrer Sicherheitskopie. Legen Sie jetzt bitte Ihre Originaldiskette in das Disklaufwerk und holen sich deren Inhaltsverzeichnis auf den Bildschirm. Mit den bekannten Möglichkeiten der Fensterbewegungen können Sie jetzt die beiden Fenster möglichst breit untereinander auf dem Bildschirm plazieren, damit Sie die Inhalte beider Fenster vergleichen können. Wenn alles geklappt hat, sollten die beiden Fenster den gleichen Inhalt zeigen.

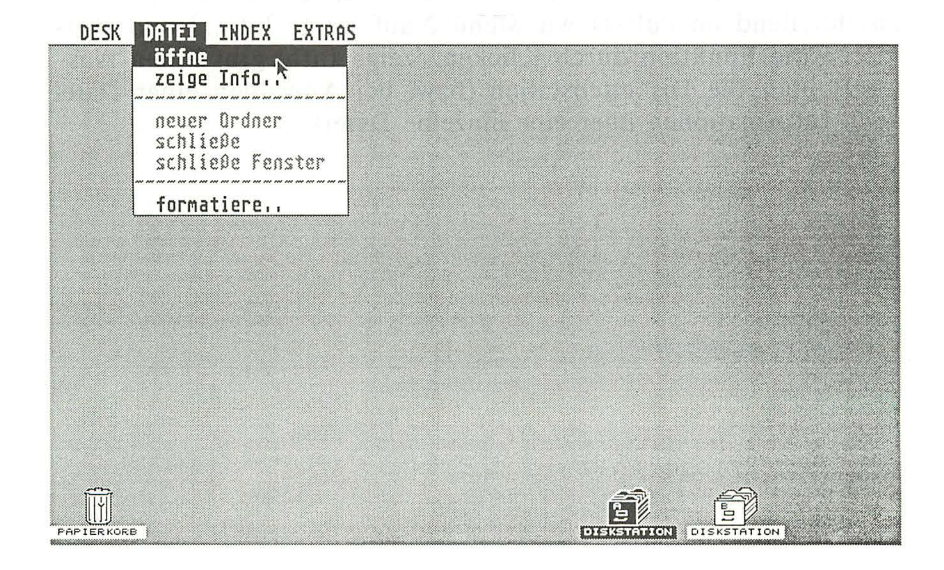
Um zu testen, ob unsere Kopie auch wirklich funktioniert, schalten Sie bitte den Computer aus, legen die Sicherheitskopie anstelle der Originaldiskette ein und verfahren genau wie im Kapitel 1.6 Startvorbereitungen beschrieben. Sollten Sie Schwierigkeiten haben, schalten Sie alle Geräte wieder aus, nehmen

Ihre Originaldiskette und probieren es damit noch einmal. Dafür haben wir diese Disketten ja noch. Sie müssen dann wohl oder übel die komplette Tortur noch einmal durchführen. Das Kopieren von Disketten werden wir im Verlauf des Buches noch an einigen Beispielen aufgreifen, damit Sie auch für jede Situation genau wissen, was wie zu machen ist. Doch jetzt wollen wir uns den anderen Pull-Down-Menüs zuwenden, denn da sind ja noch einige Dinge versteckt, die wir bisher nicht kennengelernt haben.

2.2.3.2 DATEI

Nun zu den Funktionen im zweiten Pull-Down-Menü, das die Überschrift 'DATEI' trägt.

Hierzu klicken Sie zuvor Diskettenstation A einmal an und bewegen anschließend den Pfeil mit der Maus auf das Feld 'DATEI'.



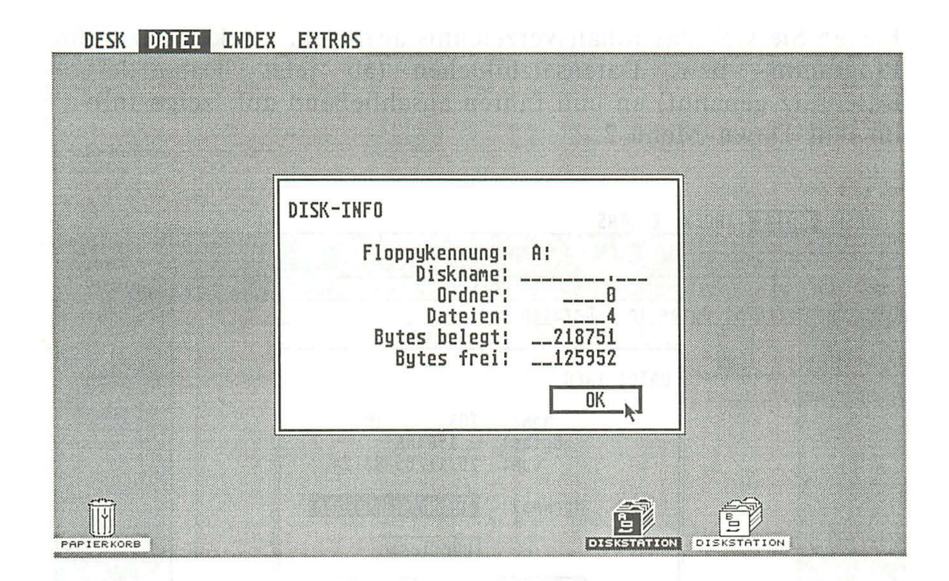
Öffne

Zuerst wird uns die Möglichkeit geboten, etwas zu 'Öffne'. Wählen Sie dieses Feld mit dem Pfeil an und drücken anschließend die linke Maustaste ... der Inhalt von Diskettenstation A wird uns angezeigt.

Warum gibt es diese Funktion, wo doch das Anzeigen der Dateien von Diskettenstation A mit einem Doppelklick die gleiche Funktion erfüllt? Eine berechtigte Frage, aber vielleicht hat ATARI bei der 'Öffne'-Funktion halt an all diejenigen gedacht, die Schwierigkeiten bei der Durchführung des Doppelklicks haben?! Also, es geht auch anders: Diskettenstation A durch einen einfachen Klick aktivieren und anschließend im Pull-Down-Menü 2 die Funktion 'Öffne' anwählen.

zeige Info...

Aktivieren Sie nun ein weiteres Mal Diskettenstation A, fahren anschließend im Pull-Down-Menü 2 auf 'zeige Info...' und aktivieren die Funktion durch Klicken. 'zeige Info' gibt Ihnen Auskunft über die Diskettenstation (bzw. bei Anwählen einer Datei auch Informationen über eine einzelne Datei).

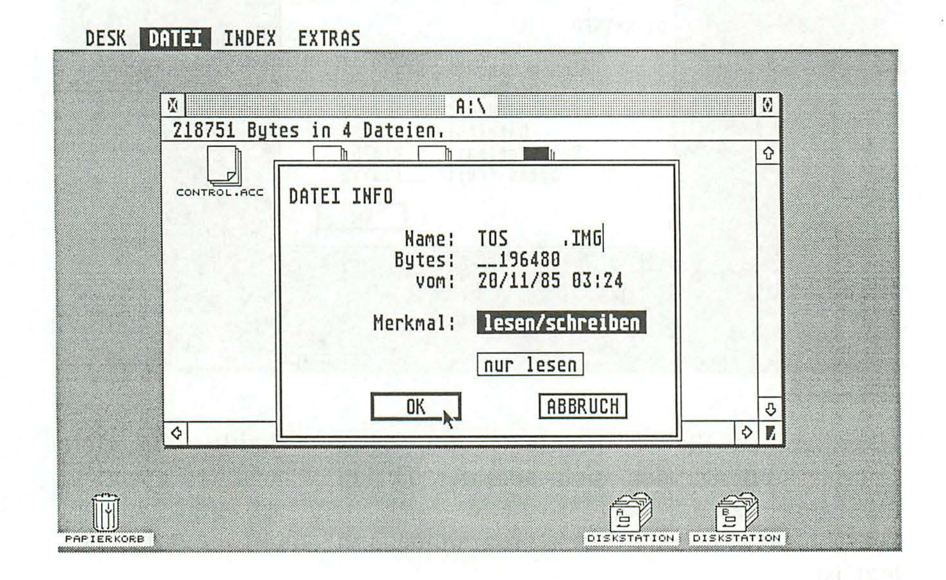


Hier können Sie erfahren, wie die Diskette heißt, wie viele Ordner und Dateien sich auf der Diskette befinden, wieviel Speicherplatz auf der Diskette noch frei ist und wieviel Speicherplatz bereits durch Programme und/oder Datensätze belegt ist.

Hier können Sie auch erfahren, ob es sich um eine ein- oder um eine doppelseitig beschriebene Diskette handelt. Wie Sie schon von der Erstellung der Sicherheitskopie wissen, hat eine einseitig beschriebene Diskette etwa 360000 freie Bytes. Auf doppelseitigen Disketten passen demnach etwa 720000 Bytes. Wenn Sie jetzt in der Disk-Info-Box die Anzahl der freien und der belegten Bytes zusammenzählen, so wird entweder ein Wert in der Größenordnung einer einseiten Disk oder einer doppelseitigen Disk herauskommen. Das die bei der Formatierung mitgeteilten freien Bytes dabei nicht genau herauskommen, soll Sie nicht stören, das hat mit dem Betriebssystem des ATARI ST zu tun und ist auch eigentlich nicht weiter wichtig.

Sie können die Informationen in diesem Menü nur zur Kenntnis nehmen, aber keine Änderungen mehr vornehmen. Anders schaut es da schon aus, wenn Sie vorher eine Datei aktivieren.

Lassen Sie sich das Inhaltsverzeichnis anzeigen, klicken darin ein Programm- bzw. Datensatzbildchen (ab jetzt "Datenfile" = Datensatz genannt) an und fahren anschließend auf 'zeige Info...' im Pull-Down-Menü 2.



Hier nun können Sie über die Tastatur den Dateinamen ändern, sowie zwischen den Funktionen 'lesen/schreiben' und 'nur lesen' auswählen. Bestätigen Sie (wie auch bei der Behandlung von Informationen über die Diskettenstation) die angezeigten Zeilen durch Ansteuern des OK-Feldes und das anschließende Anklicken dieser Funktion.

Die folgenden 3 Funktionen im Pull-Down-Menü Nr. 2 heißen 'neuer Ordner', 'schließe' und 'schließe Fenster'.

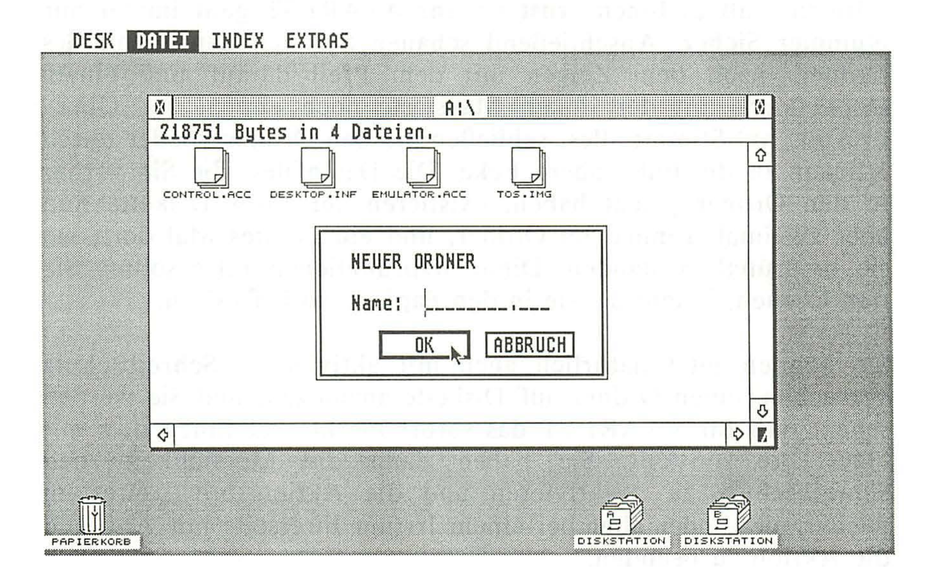
schließe und schließe Fenster

Soweit Ihnen der Aufbau eines Datenblattes (wie vorhin unser Inhaltsverzeichnis der Diskette und die Funktionen der verschiedenen Ecken) noch bekannt ist, können Sie die Funktionen 'schließe' und 'schließe Fenster' ruhig übergehen. Benutzen Sie

anstatt dessen die linke obere Ecke der Anzeigebildschirme, schon werden eifrig die Papiere wieder in die Diskettenstation/in den Karteikasten zurücksortiert, mehr nicht.

neuer Ordner

Was aber hat ein Ordner für eine Bedeutung? Wählen Sie einfach die Ordnerfunktion an und warten ab, was da passiert:



Sie werden nach einem Namen für den Ordner befragt und können anschließend die OK-Anzeige anklicken.

Bedenken Sie bitte, daß Sie bei allen Aktionen, die irgend etwas auf die Diskette schreiben, Ihre Sicherheitskopie benutzen. Damit Sie überhaupt einen neuen Ordner auf der Diskette anlegen können, darf der Schreibschutzschalter nicht geöffnet sein. Überprüfen Sie also zuerst, ob die Öffnung geschlossen ist.

Wenn Sie also das OK-Feld angeklickt haben und eine nicht schreibgeschützte Diskette verwenden, haben Sie nun einen Ordner zur Verfügung und können all die Datenfiles auf dem

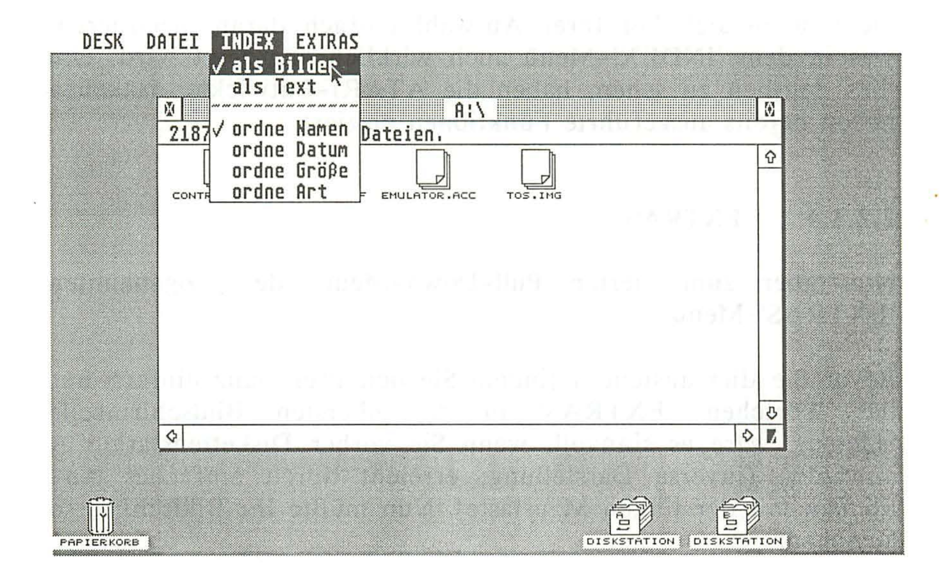
Bildschirm verschieben und - bildlich gesprochen - in diesen Ordner plazieren, wie wir dies auch vorhin bereits beim Verschieben der Gegenstände auf dem Computerschreibtisch (Diskettenstation und Papierkorb) getan haben. Schieben Sie also nun (bei gedrückter Maustaste) einige Programme in den Ordner, wie es Ihnen beliebt.

Jedesmal, wenn Sie ein Datenfile in den Ordner packen wollen, fragt Sie Ihr ST vorher, und Sie müssen durch Ihr OK bestätigen, daß es Ihnen ernst ist. Ihr ATARI ST geht immer auf Nummer Sicher. Anschließend schauen Sie sich den Inhalt des Ordners nach dem Zeigen mit dem Pfeil darauf und einem Doppelklick mit der linken Maustaste noch einmal zur Überprüfung an! Stimmt alles, schließen Sie den Ordner wieder durch Klicken in die linke obere Ecke. Die Datenfiles, die Sie vorher in den Ordner gelegt haben, existieren auf Ihrer Diskette nun aber zweimal. Einmal im Ordner, und ein zweites Mal dort, wo sie ursprünglich standen. Diese ursprünglichen Files sollten Sie nun löschen, indem Sie sie in den Papierkorb befördern.

Sie können jetzt natürlich auch mit aktiviertem Schreibschutz versuchen, einen Ordner auf Diskette anzulegen, und Sie werden sehen, daß Ihr ATARI ST das sofort merkt und Ihnen dies mit einer Box mitteilt. Sie haben dann die Möglichkeit, den Schreibschutz zu deaktivieren und die Aktion mit Weiter zu wiederholen, oder aber bei einem Irrtum Ihrerseits mit Abbruch die Aktion zu beenden.

2.2.3.3 INDEX

Schließen wir die Bilddarstellung wieder und schauen uns das Menü 3 mit dem Titel 'INDEX' etwas näher an.



INDEX als Bilder oder als Text

Hier nun können Sie auswählen (und bekommen dies auch im Disketteninhaltsverzeichnis so angezeigt), wie z.B. das Inhaltsverzeichnis auf dem Bildschirm aussehen soll ... ob nun mit Bildern oder ohne Bilder, ob nach Name oder Datum oder Größe oder Art sortiert.

Sobald Sie eine Funktion angewählt haben, wird sie auch schon ausgeführt, so daß ihre Wirkung bereits kurz nach Anklicken des entsprechenden Auswahlpunktes im angezeigten Inhaltsverzeichnis zu sehen ist.

Ordne nach Name, Datum, Größe oder Art

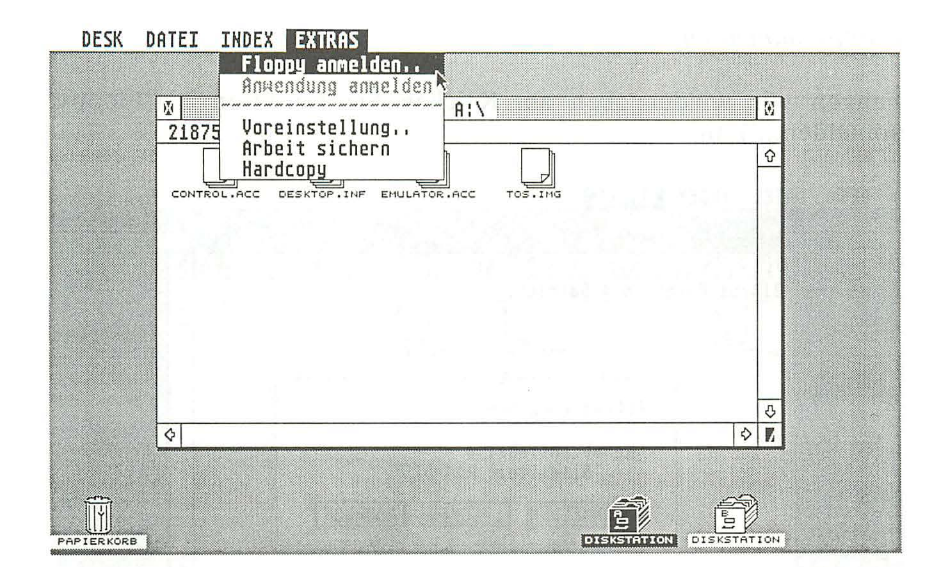
Treffen Sie selbst die richtige Wahl, ob mit Text oder Bild (den sogenannten Icons), ob nach was auch immer sortiert.

Sie können sich bei Ihrer Auswahl einfach daran orientieren, was in dem 'INDEX'-Menü auch wirklich angezeigt wird. Um dies deutlich zu sehen, haben die ATARI-Entwickler Häkchen neben bereits ausgeführte Funktionen plaziert.

2.2.3.4 EXTRAS

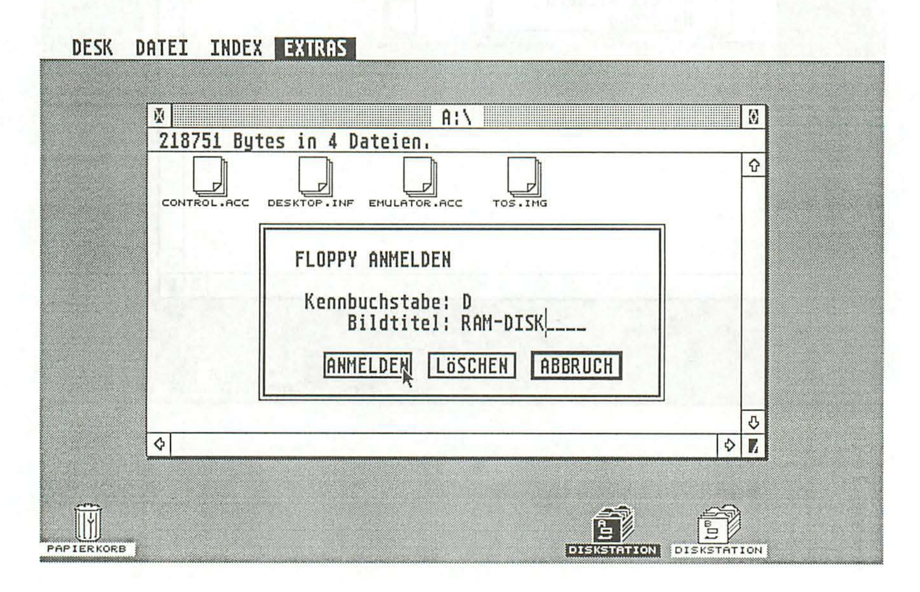
Nun aber zum vierten Pull-Down-Menü, dem sogenannten 'EXTRAS'-Menü.

Bevor Sie dies ansteuern (indem Sie den Pfeil ganz einfach auf das Wörtchen 'EXTRAS' in der obersten Bildschirmzeile richten), wäre es sinnvoll, wenn Sie vorher Diskettenstation A anwählen (inverse Darstellung, erreicht durch einfaches Anklicken mit der linken Maustaste). Nun müßte Ihr Bildschirm so aussehen:


DESK
DATEI
INDEX
EXTRAS
Floppy anmelden..
Anwendung anmelden
Voreinstellung..
Arbeit sichern
Hardcopy
A:\
21875
CONTROL.ACC
DESKTOP.INF
EMULATOR.ACC
TOS.IMG
PAPIERKORB
DISKSTATION
DISKSTATION

Floppy anmelden...

Fangen wir wieder oben an: Klicken Sie Funktion 1 ('Floppy anmelden...') an:



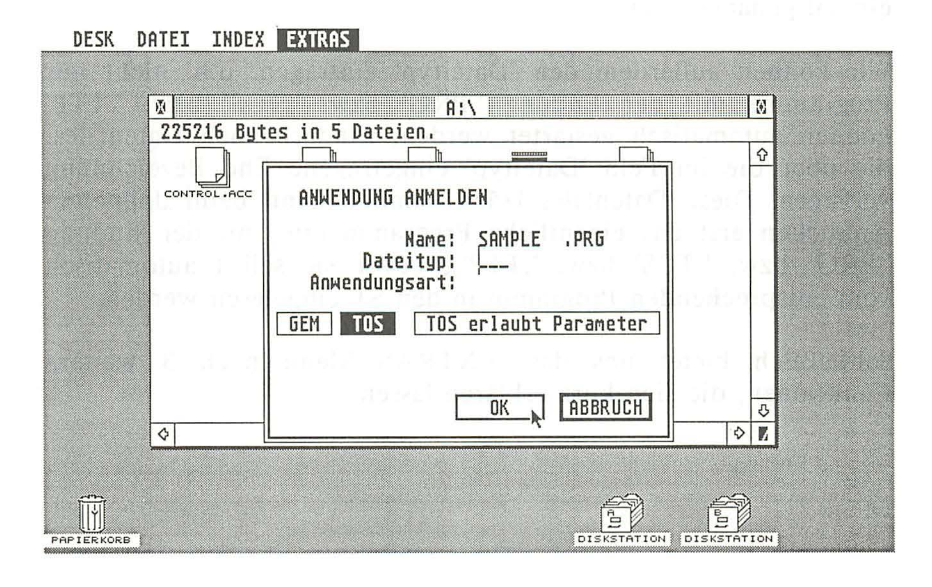
Hier nun können Sie die unter dem Bildsymbol stehende Bezeichnung (im Augenblick 'DISKSTATION') sowie die Diskettenkennzeichnung (im Augenblick 'A') verändern.

Es ist auch möglich, die gesamte Diskettenstation vom Bildschirm zu 'LÖSCHEN' (Anwählen des mittleren Feldes), jedoch seien Sie sehr vorsichtig mit dieser Funktion, denn wenn Sie erst einmal beide Diskettenstationen gelöscht haben, ist Ihre Betriebssystem-Diskette nicht mehr zu gebrauchen (vorausgesetzt, Sie haben Ihre Arbeit vorher 'gesichert').

Anwendung anmelden

Die Funktion 'Anwendung anmelden' erscheint auf dem Bildschirm zur Zeit recht schwach, und Sie werden merken, daß diese Funktion im Augenblick nicht anzuwählen ist.

Sie müssen zu diesem Zweck erst einmal ein Programm aus dem Inhaltsverzeichnis anklicken (jedes Programm trägt die Bezeichnung '.PRG' oder '.TOS' am Ende und ist durch ein Blocksymbol dargestellt) und können anschließend unter 'EXTRAS' die gewünschte Anwendung anmelden.



Was heißt das? Unser ATARI ST kann nicht nur Programme verarbeiten, die mit der Maus gesteuert werden (die Benutzerschnittstelle GEM ist dafür verantwortlich), sondern auch Programme ohne diese phantastische Grafik-Schnittstelle. Diese Programme laufen dann ohne die grafische Symbolsteuerung. GEM (Sie erinnern sich, der Teil des Betriebssystems, der für die Symbolbenutzerführung zuständig ist) wird dann praktisch "ausgeblendet", und das Programm läuft "nur" unter dem Betriebssystem TOS.

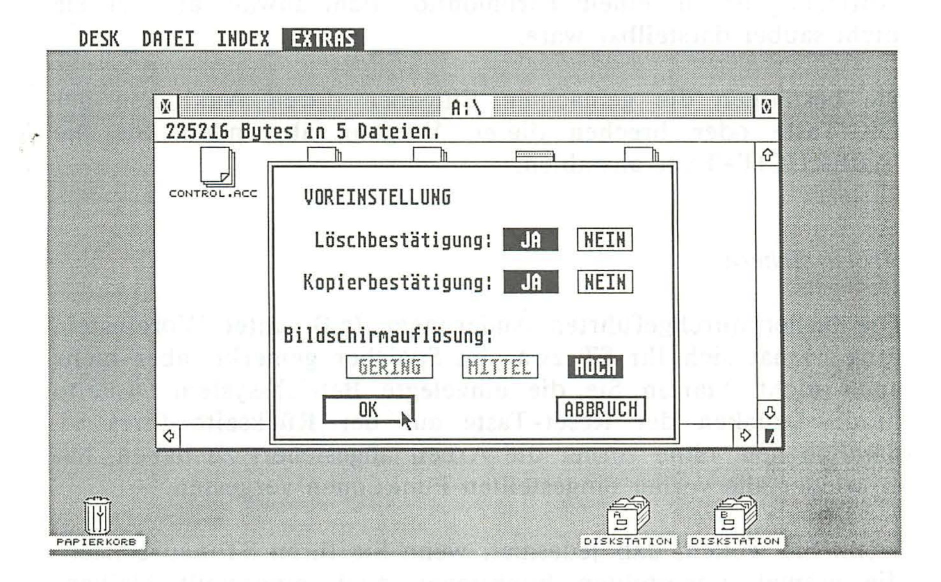
Für direkt zu startende Programme (mit '.PRG' am Ende bzw. durch die Darstellung des Blocks leicht von den anderen Bildern oder Icons zu unterscheiden), können wir mit dem Punkt 'Anwendung anmelden' auswählen, unter welchem Betriebssystem ('GEM', 'TOS' oder 'TOS übernimmt Parameter') unser Programm starten soll, was man "die Anwendungsart auswählen" nennt.

Auch wenn manches für Sie vielleicht noch ein bißchen unverständlich ist, im Laufe Ihrer Arbeit mit dem ATARI ST werden Sie verstehen, wovon wir hier reden. Dann sollten Sie hier noch einmal genauer lesen.

Wir können außerdem den 'Dateityp' eintragen, d.h. nicht nur Programme mit der Endung '.PRG' bzw. '.TOS' bzw. '.TTP' können automatisch gestartet werden, sondern auch Datenfiles, die über die im Feld 'Dateityp' eingetragene End-Bezeichnung verfügen. Diese Datenfiles laden nämlich dann beim doppelten Anklicken erst das eigentliche Programm ein (mit der Endung '.PRG' bzw. '.TOS' bzw. '.TTP'), bevor sie selbst automatisch vom entsprechenden Programm in den ST eingelesen werden.

Schließlich bietet uns das 'EXTRAS'-Menü noch 3 weitere Funktionen, die sich kurz erklären lassen:

Voreinstellung...



Wenn Sie eine Datei löschen wollen, werden Sie im Normalfall vorher extra noch einmal danach gefragt, ob Sie dies auch wirklich tun wollen (Sicherheitsabfrage). Ebenso verfährt Ihr ST, wenn Sie eine Datei oder eine ganze Diskette kopieren wollen.

Geht Ihnen diese andauernde Sicherheits-Abfragerei auf den Wecker, und sind Sie sich in der Handhabung der Lösch- bzw. Kopierfunktion wirklich vollkommen sicher, klicken Sie ganz einfach bei 'Löschbestätigung' bzw. 'Kopierbestätigung' das Feld 'NEIN' an.

Nun starten die Kopier- bzw. Löschvorgänge einen Schritt früher (das Risiko, etwas falsch zu machen, ist allerdings in dem Fall auch wesentlich größer!!!).

Besitzen Sie einen Schwarz/Weiß-Monitor, können Sie beim Einstellen der Bildschirmauflösung nur 'HOCH' anwählen, die Felder 'GERING' und 'MITTEL' sind gesperrt. Bei Anschluß eines Farbmonitors an Ihren ST ist dies genau umgekehrt: Sie

können nur geringe bzw. mittlere Auflösung anwählen; hohe Auflösung ist auf einem Farbmonitor nicht anwählbar, weil sie nicht sauber darstellbar wäre.

Sie bestätigen die gemachten Eingaben durch Anklicken der OK-Taste oder brechen diesen Vorgang ab, indem Sie die 'ABBRUCH'-Taste anwählen.

Arbeit sichern

Die bisher durchgeführten Änderungen (z.B. unter 'Voreinstellung...') hat sich Ihr ST zwar im Speicher gemerkt, aber mehr auch nicht. Starten Sie die eingelegte Betriebssystem-Diskette durch Drücken der Reset-Taste auf der Rückseite Ihres ST nämlich neu, ohne vorher die Arbeit abgesichert zu haben, hat er wieder alle vorher eingestellten Funktionen vergessen.

Wenn Sie wollen, daß jedesmal, wenn Sie Ihren ST neu starten, die einmal eingestellten Funktionen auch eingestellt bleiben, müssen Sie nach allen durchgeführten Änderungen unter 'EXTRAS' die 'Arbeit sichern'. Ihr ST legt dann auf der Diskette ein entsprechendes Datenfile mit Namen 'DESKTOP.INF' an, das er bei jedem neuen Einladen der Betriebssystemdiskette in den Speicher einliest und somit den zuvor eingestellten Zustand wiederherstellt.

Hardcopy

Zu dieser Funktion braucht man sicherlich nicht viel zu sagen, ihre Funktion ergibt sich von selbst, soweit Sie einen grafikfähigen Drucker an die Centronics-Schnittstelle (Anschluß 8 des ST, siehe Foto vorne) Ihres ST angeschlossen und zuvor die Centronics-Schnittstelle mit der Drucker-Anpassung (unter 'DESK') richtig eingestellt haben. Die Funktion bewirkt, daß Ihr Drucker Ihnen den Bildschirm so ausdruckt, wie er zur Zeit ist.

Das gleiche Ergebnis können Sie übrigens auch durch gleichzeitiges, kurzes Drücken der "Alternate"- und "Help"-Tasten erreichen. Vorteil der zweiten Möglichkeit: Das Pull-Down-Menü zum Aufrufen dieser Druckfunktion ist nicht mit auf dem Ausdruck.

Falls Sie keinen Drucker an Ihrem ST angeschlossen haben, sollten Sie kein Hardcopy aufrufen, da Sie sonst 30 Sekunden warten müssen, bis Ihr Rechner wieder reagiert.

Blitter

Wenn Sie das Blitter-TOS eingebaut haben *und* über einen eingebauten Blitter verfügen, werden Sie einen weiteren Menüpunkt 'Blitter' entdecken. Beim Anklicken wird jeweils ein Häkchen gesetzt oder gelöscht. Das bewirkt, daß einmal der sogenannte Blitter ein- bzw. ausgeschaltet ist. Mit eingeschaltetem Blitter laufen viele Funktionen schneller, z.B. der Aufbau eines Fensters für das Directory. Normalerweise können Sie den Blitter immer eingeschaltet lassen. Bei Spielen, die damit zu schnell laufen, oder Programmen, die gar nicht funktionieren wollen, kann aber durch Abschalten des Blitters eventuell Abhilfe geschaffen werden.

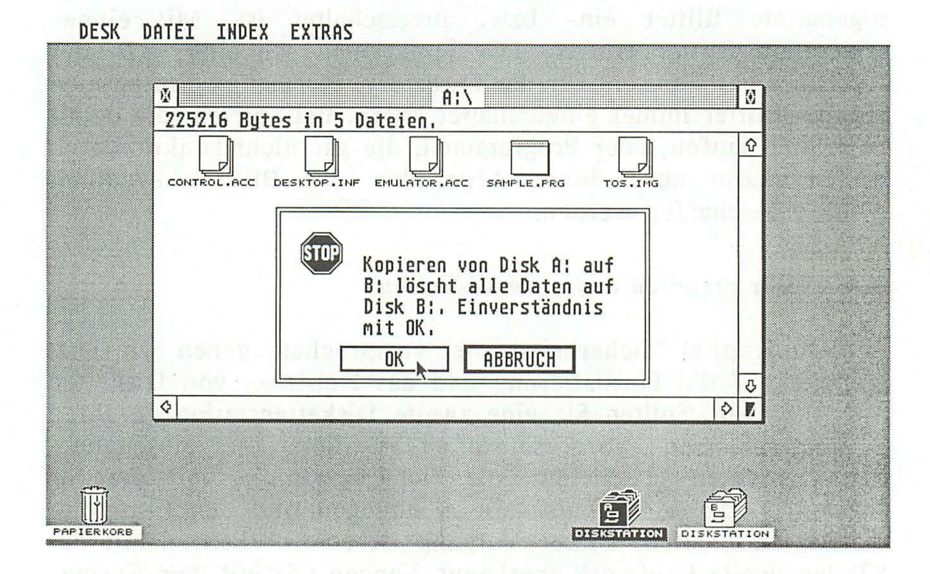
2.3 Wir erstellen uns neue Disketten

Wie im Kapitel "Sicherheitskopie" versprochen, gehen wir jetzt genauer auf die Formatierung und das Kopieren von Disketten allgemein ein. Sollten Sie eine zweite Diskettenstation an Ihren ST angeschlossen haben, so sollten Sie diese jetzt einschalten, den Computer ausschalten, die Sicherheitskopie, mit der Sie Ihren ST starten, in Laufwerk A einlegen und den Computer wieder einschalten. Dieser Vorgang ist erforderlich, damit der ST das zweite Laufwerk anerkennt. Fangen wir mit dem Formatieren an. Zu allererst legen Sie bitte die leere, neue Diskette in das Laufwerk, in dem Sie formatieren wollen. Dann müssen wir dem Computer wieder mitteilen, in welchem Disklaufwerk die zu formatierende Diskette liegt. Dies wird durch einmaliges Anklicken der entsprechenden Diskstation mit der linken Maustaste gemacht, bis diese signalisiert, daP sie verstanden hat. In unserem Fall benutzen Sie bitte Laufwerk A, damit Sie der Beschreibung genau folgen können. Starten Sie nun die Formatierfunktion, indem Sie im Pull-Down-Menü "Datei" den Punkt "formatiere.." anklicken. Bitte achten Sie auch hier wieder darauf, daß Sie wirklich eine neue Diskette oder aber eine, deren Daten Sie nicht mehr brauchen, in das Disklaufwerk A gelegt haben, denn nach dem Formatieren sind alle Daten gelöscht.

Wenn Sie unserem Tip gefolgt sind, alle Originaldisketten schreibgeschützt haben und jetzt nur mit den Sicherheitskopien arbeiten, kann Ihnen nichts passieren.

2.3.1 formatiere...

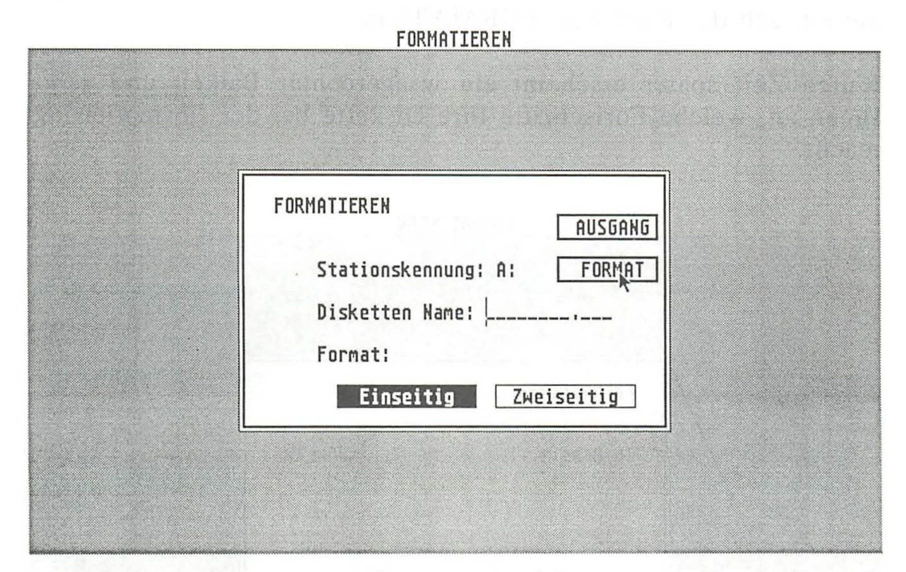
Ein Hinweisfeld macht uns nun darauf aufmerksam, welche furchtbaren Folgen die Bestätigung dieser Frage mit sich bringen kann:



Sämtliche Daten auf Ihrer Diskette würden somit gelöscht, wenn Sie den Befehl 'formatiere...' ausführen ließen.

Wofür brauchen wir die Formatier-Funktion, wenn nicht für das Löschen von Disketten? Wenn Sie sich bereits einige 3.5 Zoll-Leerdisketten zugelegt haben, haben Sie vielleicht schon bemerkt, daß diese Disketten so noch nicht für Ihren ST zu gebrauchen sind. So sehr Sie sich auch bemühen, so zärtlich wir auch die Maustaste drücken, der ST will einfach nicht. Erst müssen Sie diese Disketten auf den Gebrauch in Ihrem ST-Computer vorbereiten.

Beim Formatieren geschieht nun diese Vorbereitung. Der ATARI ST teilt die Diskette dabei in einzelne Spuren auf, auf die er hinterher Programme und Daten speichern wird. Er legt sich praktisch die Schubladen an, in die er hinterher die Daten legen wird.

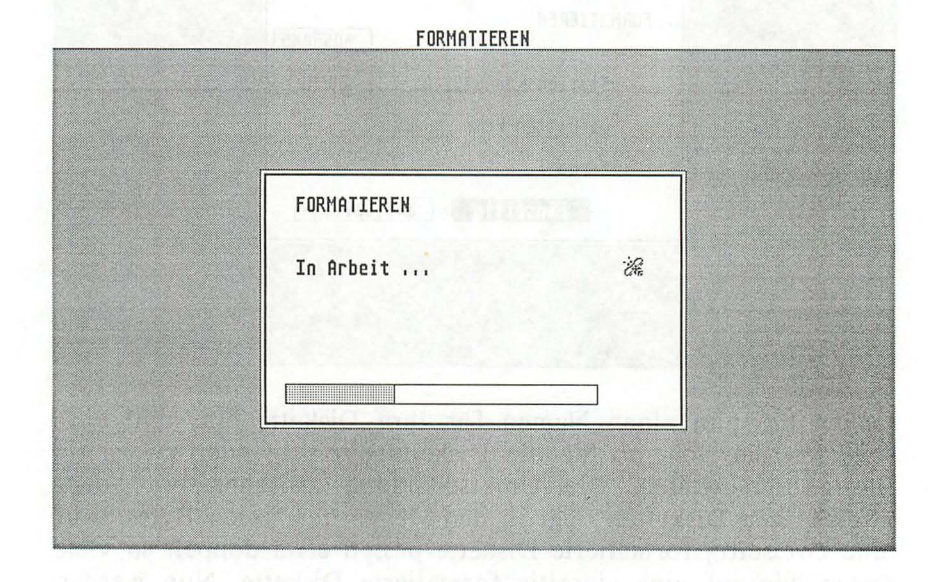


Geben Sie nun einen Namen für Ihre Diskette ein, und entscheiden Sie sich, ob Sie die Diskette einseitig oder zweiseitig formatieren wollen. Die Unterscheidung zwischen ein- oder zweiseitigen Disketten liegt in der Menge der freien Bytes. Auf eine zweiseitig formatierte Diskette passen etwa doppelt so viele Daten wie auf eine einseitig formatierte Diskette. Nun werden Sie sagen, dann formatiere ich meine Disketten alle zweiseitig. Dazu müssen Sie jedoch wissen, daß das nicht mit jedem Disklaufwerk möglich ist. Sind Sie Besitzer einer Diskstation SF 354, so können Sie Ihre Disketten grundsätzlich nur einseitig formatieren. Bei den Laufwerken mit der Bezeichnung SF 314 und denen, die im ATARI 1040 STF eingebaut sind, können Sie wahlweise ein- oder zweiseitig formatieren. Achten Sie bei Ihrer Diskettenwahl dann aber bitte auch darauf, daß Sie sich zweiseitig beschreibbare Disketten besorgen. Zwar lassen sich die billigeren Disketten mit der Bezeichnung "Einseitig" auch zweiseitig formatieren, aber Sie müssen immer mit Datenverlust

rechnen, da der Hersteller der Diskette die zweite Seite nicht auf Fabrikationsmängel geprüft hat und auf diese Seite daher auch keine Gewähr auf Datensicherheit ergibt.

Nachdem Sie also das Entsprechende angewählt haben, klicken Sie einfach das Kästchen 'FORMAT' an.

Kurze Zeit später erscheint ein waagerechter Balken und zeigt Ihnen an, welche Fortschritte Ihre Diskette bei der Formatierung macht.



Schließlich ist der Vorgang abgeschlossen, und Sie erfahren wiederum durch ein angezeigtes Fenster, daß die Diskette einwandfrei formatiert wurde, d.h., daß z.B. zum Gebrauch bei einer einseitigen Diskette '357376 Bytes' oder Speicherplätze zur freien Verfügung stehen.

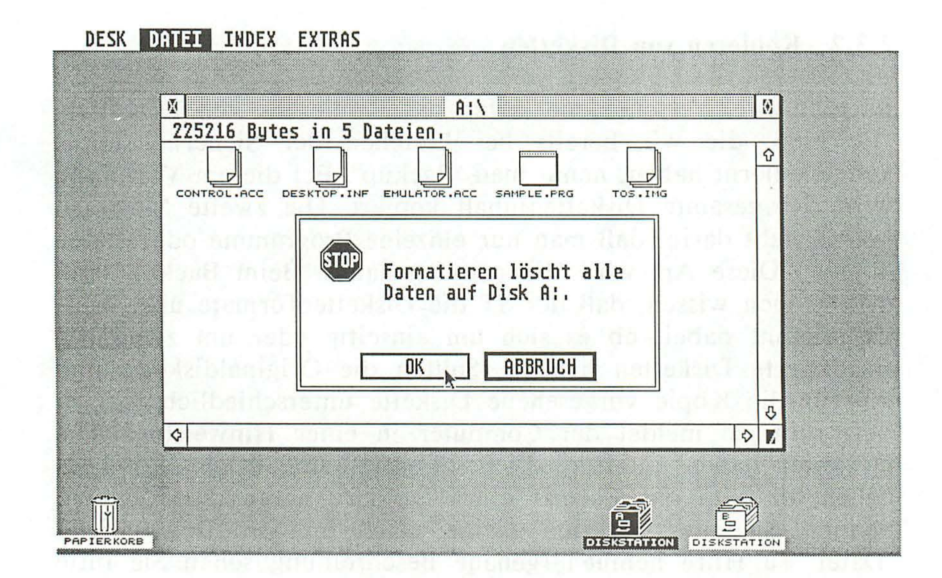
Sie können nun weitere Disketten formatieren oder diese Funktion durch Anklicken des 'AUSGANG'-Feldes wieder verlassen.

2.3.2 Kopieren von Disketten

Es gibt zwei verschiedene Arten des Kopierens von Disketten. Die eine, die wir bereits bei Erstellen der Sicherheitskopie kennengelernt haben, nennt man "Backup". Bei diesem Verfahren wird der gesamte Disketteninhalt kopiert. Die zweite Möglichkeit besteht darin, daß man nur einzelne Programme oder Daten kopiert. Diese Art wird "Filecopy" genannt. Beim Backup muß man jedoch wissen, daß der ST die Diskettenformate überprüft. Er erkennt dabei, ob es sich um einseitig oder um zweiseitig formatierte Disketten handelt. Sollten die Originaldiskette und die für die Kopie vorgesehene Diskette unterschiedlich formatiert sein, so meldet der Computer in einer Hinweisbox "Die Disketten haben unterschiedliches Format" und bricht das Vorhaben ab. Um das Format einer Diskette herauszubekommen, können Sie die Funktion "zeige Info" im Pull-Down-Menü "Datei" zu Hilfe nehmen (genaue Beschreibung sehen Sie bitte unter der Beschreibung des Menüpunktes "zeige Info" nach).

Zuerst wollen wir uns noch einmal mit "Backup" beschäftigen. Da Sie bisher noch keine andere Diskette haben, nehmen Sie noch einmal die Language-Disk und kopieren diese. Sie sollten auch dieses Beispiel nachvollziehen, da wir mit der Kopie noch einige Dinge ausprobieren wollen. Denken Sie daran, daß Sie jetzt nur eine einseitig formatierte Diskette gebrauchen können, da die Language-Disk auch nur einseitig formatiert ist.

So, nun geht es endlich an das Kopieren. Sie wählen das Bild von Diskettenstation A mit dem Pfeil durch einfaches Anklicken an und verschieben es über Diskettenstation B bis das Ikon schwarz wird. Wieder werden Sie auf eventuelle Vorsichtsmaßnahmen hingewiesen, bevor der Kopiervorgang (auf eine vorher formatierte Diskette) beginnen kann.



Besitzen Sie zwei Diskettenstationen, ist der Kopiervorgang sehr einfach: Sie legen die Originaldiskette (in unserem Fall die Language-Disk) in Laufwerk A und die Diskette, auf die kopiert werden soll, in Laufwerk B ein und bestätigen schließlich diesen Vorgang durch Anklicken des 'KOPIEREN'-Feldes.

Benutzen Sie "nur" eine Diskettenstation, werden Sie nach und nach dazu aufgefordert, die Disketten (Original = A, Kopie = B) in Laufwerk A zu wechseln. Ihr ST-Computer lädt sich große Teile des Disketteninhalts vom Original in den Speicher ein, bevor er diese erneut auf die noch leere, aber bereits zuvor (s.o.) formatierte Diskette wieder ausgibt.

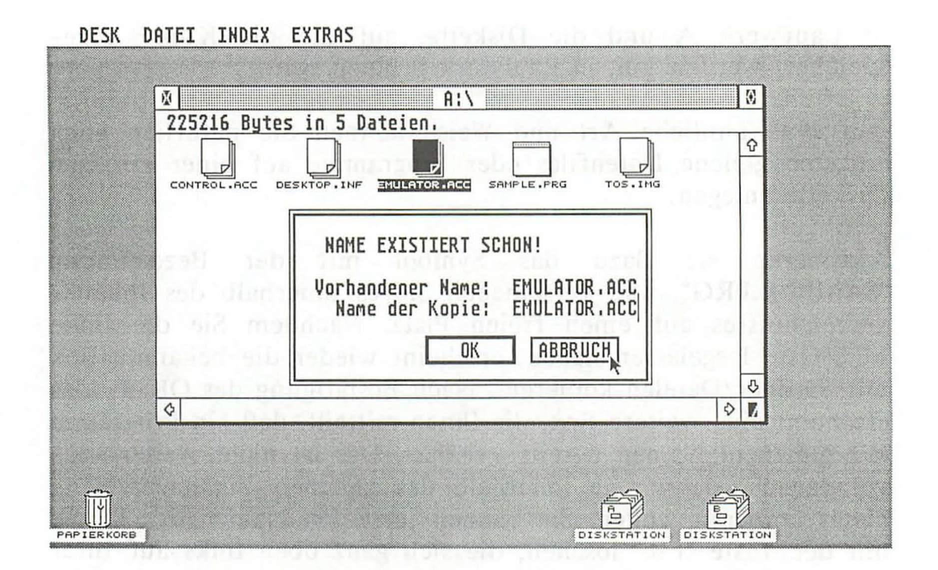
2.4 Kopieren und Löschen von Programmen und Dateien

Das Kopieren einzelner Datenfiles bzw. Programme nehmen Sie so vor, indem Sie vom angezeigten Inhaltsverzeichnis ein Bildchen anklicken und es auf das Bild von Diskettenstation B oder auf deren Inhaltsverzeichnis plazieren. Nach einer sehr ähnlichen Anzeigeprozedur wie beim Kopieren von ganzen Disketten, werden Sie nun nach und nach dazu aufgefordert, das Original

in Laufwerk A und die Diskette, auf die die Kopie abgespeichert werden soll, in Laufwerk B einzulegen.

Auf eine ähnliche Art und Weise können Sie natürlich auch einzelne gleiche Datenfiles oder Programme auf einer einzigen Diskette anlegen.

Aktivieren sie dazu das Symbol mit der Bezeichnung "SAMPLE.PRG", und verschieben Sie es innerhalb des Inhaltsverzeichnisses auf einen freien Platz. Nachdem Sie die linke Maustaste losgelassen haben, erscheint wieder die bekannte Box mit "Ordner/Dateien kopieren". Nach Bestätigung des OK-Feldes erscheint eine weitere Box, die Ihnen mitteilt, daß ein Programm mit gleichem Namen bereits existiert. Das ist nicht weiter verwunderlich, da wir ja innerhalb des gleichen Inhaltsverzeichnisses kopieren wollen. Sie können jetzt den Namen der Kopie mit der Taste "Esc" löschen, die sich ganz oben links auf Ihrer Tastatur befindet, und einen neuen Namen eingeben, den Sie mit der 'RETURN'-Taste oder durch Anklicken des OK-Feldes übernehmen. Nennen Sie die Kopie bitte "SAMPLE1.PRG". Wenn Sie sich nun das Inhaltsverzeichnis anschauen, werden Sie sowohl "SAMPLE.PRG" als auch "SAMPLE1.PRG" finden. So, jetzt wollen wir einmal zwei Programme gleichzeitig kopieren. Dazu ziehen wir einen Rahmen um die beiden Programme "SAMPLE.PRG" und "SAMPLE1.PRG", indem wir den Mauspfeil etwas links neben das erste zu kennzeichnende Symbol plazieren, die linke Maustaste drücken und festhalten, den Mauspfeil etwas rechts neben das letzte zu kennzeichnende Symbol bewegen und die Maustaste loslassen. Nun sollten Sich die beiden Symbole gefärbt haben. Wenn wir jetzt ein Symbol verschieben, so bewegt sich das zweite gleich mit. Probieren Sie es einmal und verschieben beide Symbole gleichzeitig auf einen freien Platz im Inhaltsverzeichnis. Wieder erscheint die Kopierbox, und nach Bestätigung erscheint wieder die Frage nach dem neuen Namen des Programms. Geben Sie hier jetzt bitte "SAMPLE2.PRG" und bei der nächsten Anfrage "SAMPLE3.PRG" ein. Wir haben jetzt viermal das gleiche Programm nur mit anderem Namen auf der Diskette.



Schließlich können Sie beim Löschen von Dateien mit Hilfe des Papierkorbs ähnlich vorgehen: Sie lassen sich den Inhalt der Diskette anzeigen und verschieben das nicht mehr gebrauchte Programm bzw. das nicht mehr gebrauchte Datenfile 'in' den Papierporb (in Wirklichkeit legen Sie das Bild des Datenfiles auf den Papierkorb). Löschen Sie jetzt bitte das Programm "SAMPLE1.PRG" von der Diskette.

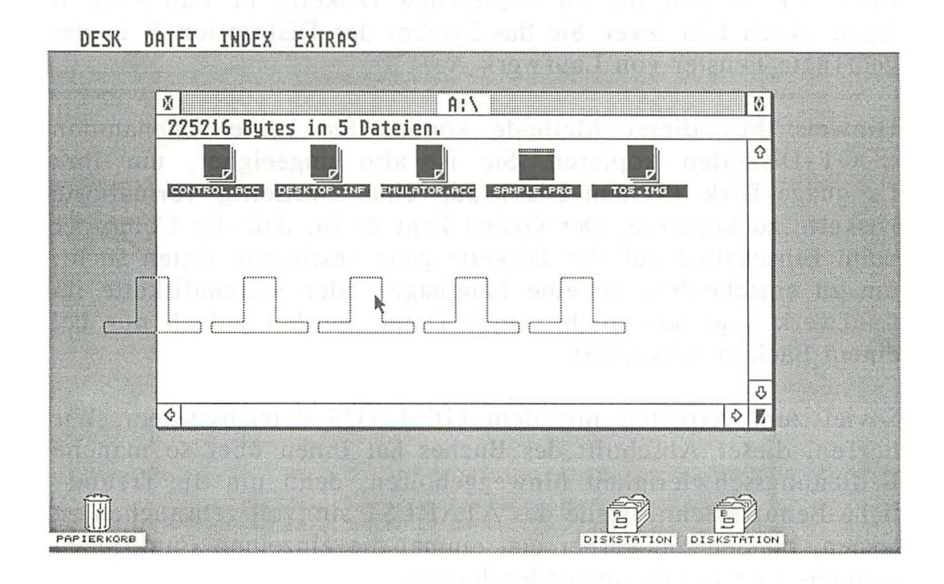
Falls Sie die Sicherheitsabfrage vorher nicht ausgestellt haben, werden Sie vor der Mächtigkeit dieser Funktion noch einmal gewarnt, andernfalls wird sofort die Löschung in die Wege geleitet.

Noch ein Wort zu dem Papierkorb. Bei Ihnen zu Hause können Sie, wenn Sie etwas in Ihren Papierkorb geworfen haben, jederzeit anfangen, in Ihrem Papierkorb herumzuwühlen und ein vorher hineingeworfenes Stück Papier wieder herausfischen. Dies geht beim ATARI-Papierkorb nicht. Was einmal im Papierkorb liegt, bleibt für immer verschwunden. In dieser Beziehung ist der Papierkorb eigentlich mehr ein Reißwolf, wie auch ATARI in der Bedienungsanleitung feststellt. Also Vorsicht bei Papierkorb-Arbeiten!

Können wir nur einzelne Programme bzw. Datenfiles löschen bzw. kopieren, oder geht dies auch gleichzeitig mit einer größeren Anzahl?

Natürlich können Sie auch gleichzeitig mehrere Files löschen. Zu diesem Zweck müssen Sie, durch Andrücken und Festhalten der Maustaste (links oben beginnend), einen Rahmen um die zu kopierenden bzw. die zu löschenden Files ziehen (ähnlich wie mit unserer Gummiecke in der rechten unteren Ecke unseres Anzeigefensters).

Wenn Sie nun anschließend einen dieser Files zum Kopieren bzw. Löschen fortbewegen wollen, bewegt sich der gesamte Pulk vorher eingerahmter Files mit. Versuchen Sie das einmal mit den Programmen "SAMPLE2.PRG" und "SAMPLE3.PRG". In unserer Abbildung sehen Sie, daß man auch mehr als zwei Symbole löschen oder kopieren kann.



Sie können aber auch bei gleichzeitig gedrückter SHIFT-Taste die gewünschten Files nacheinander auswählen. Also SHIFT-Taste gedrückt halten und nacheinander bestimmte Datenfiles

anklicken. Auch hier bewegen sich alle Files mit, soweit Sie einen von ihnen anwählen und fortbewegen wollen (die Funktion des Anwählens mit Gummirahmen bzw. des aufeinanderfolgenden Anklickens bei gedrückter SHIFT-Taste läßt sich übrigens auch auf das Anlegen von Ordnern anwenden).

Und nun ein Tip. Sollten Sie von einem Freund eine einseitig beschriebene Diskette bekommen, die Sie sich kopieren wollen, so wäre es ja eine Verschwendung, wenn Sie Ihre doppelseitigen Disketten auch nur auf eine Seite beschreiben würden. Da aber ein Backup mit unterschiedlichen Formaten nicht funktioniert, müssen wir den Umweg über das Filecopy machen. Formatieren Sie zunächst Ihre Diskette zweiseitig und lassen sich den Disketten-Inhalt davon auf Laufwerk A anzeigen. Legen Sie das Symbol der Diskstation B in das geöffnete Fenster und befolgen Sie bitte die Anweisungen des Computers. Sofern Sie über zwei Laufwerke verfügen, müssen Sie die formatierte Diskette in Laufwerk A und die zu kopierende Diskette in Laufwerk B legen. Auch hier legen Sie das Symbol der Diskstation B in das geöffnete Fenster von Laufwerk A.

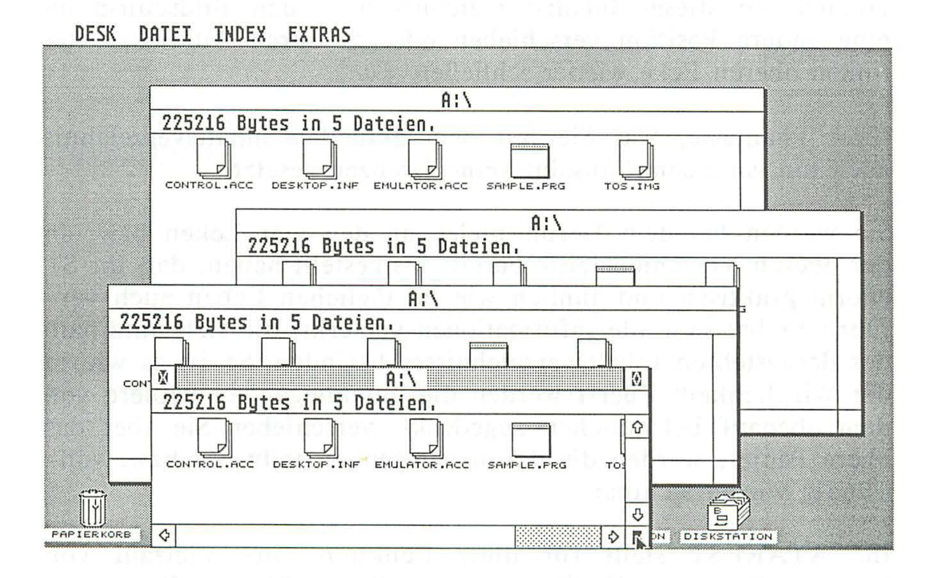
Hinweis: Mit dieser Methode können Sie keine sogenannten BOOT-Disketten kopieren. Sie ist also ungeeignet, um Ihre Language-Disk (System-Disk) auf eine zweiseitig formatierte Diskette zu kopieren. Der Grund liegt darin, daß der Computer beim Einschalten auf der Diskette ganz bestimmte Daten sucht, um zu entscheiden, ob eine Language- oder Systemdiskette im Laufwerk liegt oder nicht. Diese Daten werden jedoch nur bei einem Backup mitkopiert.

Soviel zum Arbeiten mit dem GEM-TOS-Betriebssystem. Wir hoffen, dieser Abschnitt des Buches hat Ihnen über so manche Bedienungsschwierigkeit hinweggeholfen, denn um die freundliche Benutzerschnittstelle des ATARI ST sinnvoll gebrauchen zu lernen, müssen Sie vorher erst einmal die einzelnen Funktionen kennen- und richtig anwenden lernen.

2.5 Der ST kann eifrig fensterln!

Bisher haben wir nur ein Fenster auf dem Bildschirm geöffnet, z.B. um das Inhaltsverzeichnis der aktuellen Diskette auf dem Bildschirm anzuzeigen. Das GEM-Betriebssystem, das Sie vorhin von Diskette in Ihren ST eingelesen haben, ist aber dazu fähig, bis zu vier verschiedene Fenster gleichzeitig darzustellen.

Um dies zu bewirken, bitten wir Sie, nacheinander viermal die Diskettenstation A durch Doppelklick dazu zu bewegen, das Inhaltsverzeichnis auszugeben. Sie sollen wohlgemerkt nicht jedesmal das angezeigte Fenster durch Anklicken der linken oberen Ecke wieder schließen, bevor Sie das nächste Fenster öffnen, sondern tatsächlich viermal nacheinander auch vier Bildschirme übereinanderlegen.



Nun sieht Ihr Bildschirm/Schreibtisch aber übervoll aus. Wenn Sie genau hinsehen, erkennen Sie, daß Ihr ST das Hinblättern von vier Inhaltsverzeichnissen ebenso vollzogen hat, als wenn Sie auf Ihrem Schreibtisch vier Blätter teilweise übereinandergelegt hätten.

Sie können das für den Computer zur Zeit aktuelle Blatt der vier angezeigten daran erkennen, daß hier sämtliche Informationen am Rand zu erkennen sind (z.B. Schließecke usw.), während die übrigen drei Blätter 'randlos' erscheinen (zwar sind diese ebenfalls durch eine Strichlinie von den anderen Inhaltsverzeichnisdarstellungen abgegrenzt, jedoch ist der eigentliche Randbereich, erkennbar an den vier Funktionsecken dort nicht sichtbar).

Die zuletzt auf dem Bildschirm ausgegebene Inhaltsverzeichnisanzeige ist also die aktuelle und auch voll funktionstüchtige. Sie können hier den Randbereich genauso benutzen, wie Sie dies auch bereits vorhin bei nur einem dargestellten Inhaltsverzeichnis ausprobiert haben. So ist es uns z.B. möglich, das Inhaltsverzeichnis durch Anklicken der rechten unteren Ecke (Gummiecke) auseinanderzuziehen oder zu stauchen. Ebenfalls können wir dieses Inhaltsverzeichnis über den Bildschirm an eine andere Position verschieben oder es durch Anklicken der linken oberen Ecke wieder schließen.

Ihrer Phantasie, was Sie mit dem aktuellen Inhaltsverzeichnis auch tun, sind somit absolut keine Grenzen gesetzt.

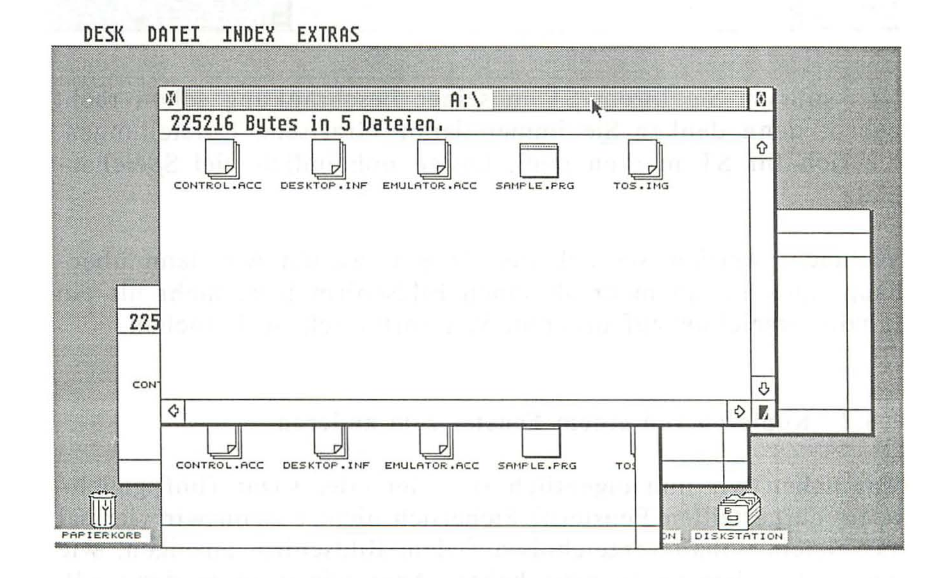
Sie werden bei dem Herumspielen an den vier Ecken bzw. an der oberen Verschiebeleiste bereits festgestellt haben, daß Ihr ST enorm praktisch und ähnlich wie im täglichen Leben auch verfährt: Er hat sich alle Informationen gemerkt, die sich unterhalb des dargestellten Inhaltsverzeichnisses befinden. So ist es wie in der Wirklichkeit: Zuerst werden die untenliegenden Papiere von dem obenauf befindlichen zugedeckt, verschieben Sie aber das obere Papier, werden die darunterliegenden teilweise bzw. vollständig wieder sichtbar.

Ihr ATARI ST stellt für diese Funktion eine Vielzahl von Speicherplätzen zur Verfügung, denn dies ist für ihn die einzige Möglichkeit, sich all die grafischen Einzelheiten im Detail zu merken. Er merkt sich nicht nur, wie das Blatt Papier aussieht, das sich direkt unterhalb des aktuellen Inhaltsverzeichnisses befindet, er merkt sich auch die Darstellung von bis zu drei weiteren Inhaltsverzeichnissen.

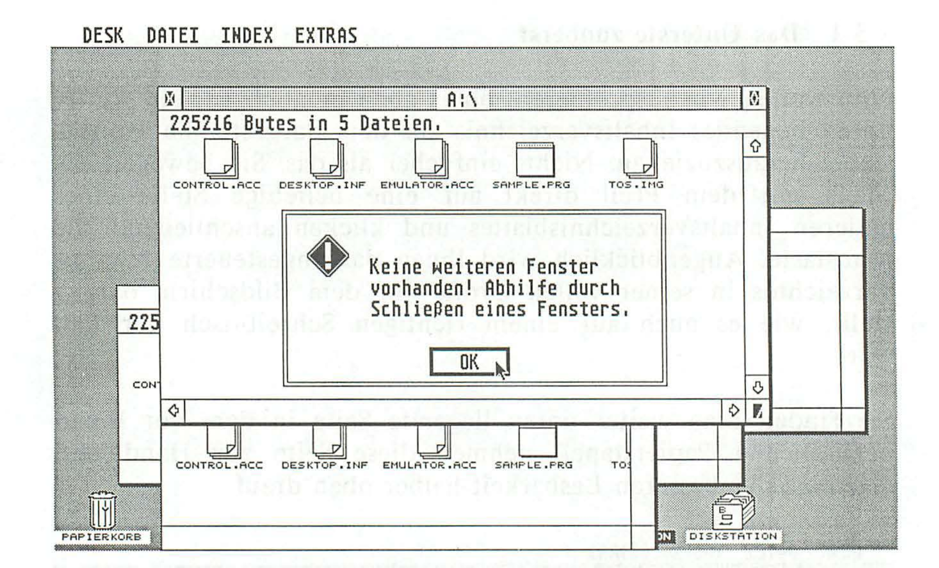
2.5.1 Das Unterste zuoberst

Nun wollen wir ausprobieren, ob es auch möglich ist, ein weiter unten liegendes Inhaltsverzeichnis aus dem dargestellten Papierstapel herauszuziehen. Nichts einfacher als das: Sie bewegen die Maus mit dem Pfeil direkt auf eine beliebige Stelle eines anderen Inhaltsverzeichnisblattes und klicken anschließend die Maustaste. Augenblicklich wird Ihnen das angesteuerte Inhaltsverzeichnis in seiner vollen Größe auf dem Bildschirm dargestellt, wie es auch auf einem richtigen Schreibtisch der Fall wäre.

Sie finden eine weiter unten liegende Seite in dem vor Ihnen befindlichen Papierstapel, nehmen diese Seite zur Hand und legen sie der besseren Lesbarkeit halber oben drauf.



Nicht anders arbeitet unser ST, egal, was sich gerade auf der Diskette befindet. Nachdem Sie vier Fenster auf dem Bildschirm dargestellt haben, werden Sie beim Öffnen eines fünften Fensters eine Meldung erhalten, daß er nun leider passen muß.



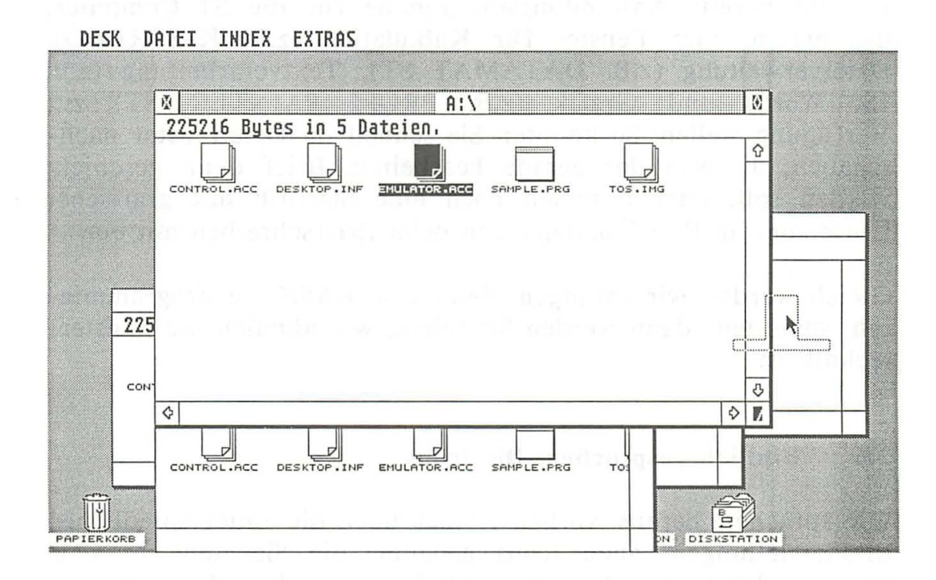
Hier müssen Sie Ihrem ST in seiner Beschränkung schon recht geben, denn denken Sie immer daran: Grafische Darstellungen, die sich Ihr ST merken muß, kosten unheimlich viel Speicherplatz.

Vielleicht werden Sie sich nun fragen, warum wir denn überhaupt gleichzeitig mehr als einen Bildschirm bzw. mehr als ein Inhaltsverzeichnis auf unserem Monitorbildschirm brauchen.

2.5.2 Kopieren von einem Fenster zum anderen

Was haben wir nun eigentlich von vier oder sogar fünf gleichzeitig dargestellten Fenstern? Sicherlich nichts, wenn wir viermal das gleiche Inhaltsverzeichnis auf dem Bildschirm anzeigen, wie wir dies soeben ausprobiert haben. Aber wie wäre es denn z.B. mit folgendem Fall: Sie legen eine Diskette ins Diskettenlaufwerk ein und lassen sich durch Doppelklick deren Inhaltsverzeichnis anzeigen. Nun stecken Sie eine andere Diskette ein und lassen sich auch deren Inhaltsverzeichnis auf dem Bildschirm anzeigen.

Vielleicht haben Sie auf der zweiten Diskette ein Programm oder Daten abgespeichert, die Sie eigentlich auf der zuerst eingelegten Diskette dringend benötigen. Kein Problem: Sie verschieben die Datei durch Festhalten des Mausknopfes ganz einfach auf die Inhaltsverzeichnisdarstellung Ihrer ersten Diskette.



Voraussetzung für diese Aktion ist es allerdings, daß Sie das Inhaltsverzeichnis zu Beginn durch das doppelte Anklicken von Diskettenstation A und das zweite Inhaltsverzeichnis durch Anklicken von Diskettenstation B erzeugt haben.

So weiß sich Ihr ST zu orientieren, von welcher Diskettenstation auf welche Diskettenstation kopiert werden soll und fordert Sie im Nachhinein dazu auf, mehrmals die Disketten im Laufwerk A zu tauschen. Der ST stellt sich nämlich vor, selbst wenn Sie nur ein Diskettenlaufwerk haben, es wären zwei vorhanden, damit er nicht durcheinander kommt. Er tut dann so, als wenn er von einem Laufwerk auf das andere kopieren würde.

Besitzen Sie allerdings tatsächlich zwei Diskettenlaufwerke, ist das ständige Wechseln der Disketten nicht nötig (fast wie ein Disk-Jockey kommt man sich schon vor, nicht wahr?). Sie legen

dann ganz einfach die Diskette mit der Originaldatei in Laufwerk A und die Diskette, auf welche die Kopie abgespeichert werden soll, in Laufwerk B. Alles weitere führt Ihr ST automatisch durch.

Es gibt bereits Anwendungsprogramme für die ST-Computer, die bis zu vier Fenster für Kalkulation (z.B. K-SPREAD), Dateiverwaltung (z.B. DATAMAT ST), Textverarbeitung (z.B. 1St Word) und Grafik (z.B. PROFI PAINTER ST) zur Verfügung stellen: So könnten Sie schnell im Dateifenster nachschauen, an wen der gerade bearbeitete Brief denn geschickt werden soll, oder beziehen noch eine Statistik mit grafischer Umsetzung in Ihre Überlegungen beim Briefschreiben mit ein.

Gleich werden wir anfangen, den ST in BASIC zu programmieren, spätestens dann werden Sie sehen, wie nützlich die Fenstertechnik ist.

2.6 Bildlich gesprochen: Die Icons

Wir sprachen bereits vorher einmal über die unterschiedlichen Bilddarstellungen, auch Icons genannt, die Sie nun auf dem Monitorbildschirm sehen. Es gibt 3 verschiedene Darstellungstypen: Den Block mit Abreißkante, den Papierstapel mit eingeknickter Ecke und den Ordner.

2.6.1 Ordner

Während wir schon auf die Arbeitsweise von Ordnern eingegangen sind, müssen wir uns hier noch mit den Blocks mit Abreißkante und Papierstapeln mit eingeknickter Ecke auseinandersetzen.

2.6.2 Block mit Abreißkante

Die Darstellung eines Blocks in unserem Inhaltsverzeichnis trägt im Namen entweder die Abkürzung 'PRG' (Programm), 'TOS' (Tramiel Operating System) oder 'TTP' (TOS Takes Parameters). Dies bedeutet, daß es sich hier um ein Programm handelt, das entweder unter dem Betriebssystem GEM (mit bis zu 4 verschiedenen Fenstern und Maussteuerung) seine Funktion ausführt (Endabkürzung 'PRG') oder aber ohne dieses Hilfsmittel auskommt (Endabkürzung 'TOS' bzw. 'TTP').

Sie können Programme direkt mit dem Mauspfeil ansteuern und drücken hiernach lediglich zweimal hintereinander den linken Mausknopf (Sie führen einen Doppelklick aus).

Nachdem sich die angeklickte Programmdarstellung auf dem Bildschirm schwarz gefärbt hat, verwandelt sich daran anschließend die Pfeildarstellung in eine geschäftige Biene (engl.: busy bee) und der Bildschirm wird gelöscht.

Nach einigen Sekunden oder bei großen Programmen vielleicht auch erst nach einer Minute, beginnt das geladene Programm zu starten (erkennbar z.B. durch eine andere Menüleiste in der ersten Bildschirmzeile).

Auf der Diskette befindet sich das Programm "SAMPLE.PRG". Nachdem Sie es gestartet haben, erscheint nach kurzer Zeit ein Fenster, in dem sich eine mit ATARI-Zeichen gefüllte Ellipse befindet. Das Fenster können Sie jetzt beliebig verkleinern und verschieben, wie Sie es bereits mit anderen Fenstern kennenge-

lernt haben. Um das Programm zu beenden, klicken Sie bitte das Schließfeld an (das ist das Feld mit dem X links oben im Fenster).

2.6.3 Papierstapel mit eingeknickter Ecke

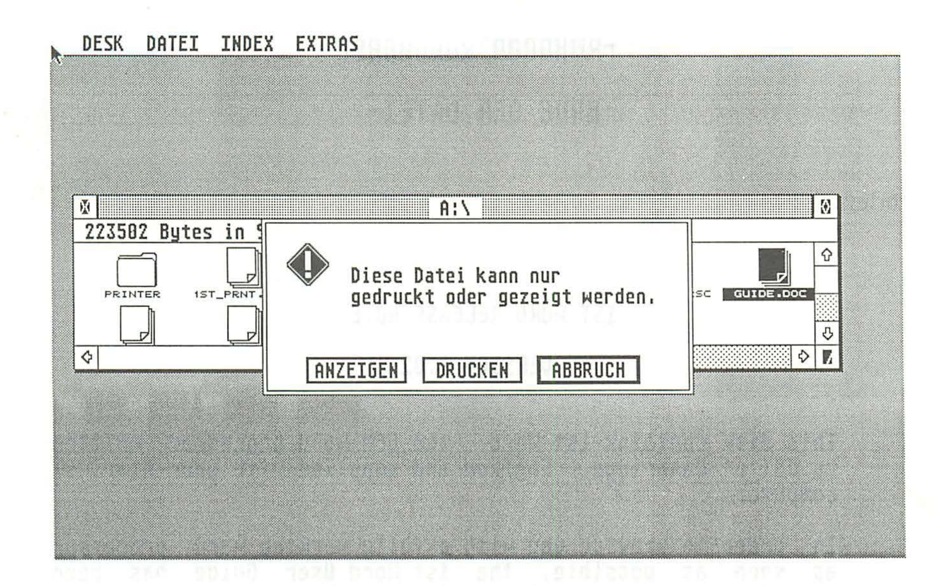
Während die Blockdarstellung ein Programm repräsentiert, handelt es sich bei dem Bildsymbol 'Papierstapel mit eingeknickter Ecke' um eine Datei.

Was ist eine Datei? Eine Datei kann sowohl ein BASIC- oder LOGO-Programm sein (mit 'Programm' ist hier nicht ein unter dem ATARI-Betriebssystem sondern z.B. ein unter der Programmiersprache BASIC startbares Programm gemeint), aber es kann dies auch ein mit einer Textverarbeitung (z.B. TEXTOMAT ST) abgespeicherter Text sein oder eine Anzahl von Adressen, die wir mit einem Dateiverwaltungsprogramm (z.B. DB Master) zuvor abgespeichert haben.

Es ist aber auch möglich, daß es sich bei Dateien um komplexe Unterprogramme handelt, die z.B. in Maschinensprache geschrieben sind, und von einem Hauptprogramm (Blockdarstellung) beim Starten automatisch nachgeladen werden. Schließlich ist es sogar möglich, daß in einer Datei das Aussehen eines Bildschirms abgespeichert ist (z.B. bei Verwendung eines Zeichenprogramms wie DEGAS). Letztgenannte Dateien werden häufig mit der Abkürzung 'PIC' (= Picture = Bild) abgespeichert und können nur durch das zur Erstellung und Abspeicherung benutzte Zeichenprogramm wieder zur Darstellung gebracht werden.

Suchen Sie sich einen 'Papierstapel mit eingeknickter Ecke' auf Ihrer BASIC-Diskette heraus (soweit dieser im Dateinamen die Abkürzung 'BAS' trägt, handelt es sich mit hoher Wahrscheinlichkeit um ein BASIC-Programm) und klicken diesen nun doppelt mit der Maus an.

Ihr ST wird augenblicklich reagieren und zeigt Ihnen nun folgende Information am Bildschirm an.



Ihr ST hat also selbständig bemerkt, daß es sich bei der doppelt angeklickten Datei um kein automatisch zu startendes Programm handelt, und bietet Ihnen nun drei Möglichkeiten an, was Sie tun können.

Sie können die getätigte Handlung (Doppelklick bei einer Datei) wieder rückgängig machen, indem Sie das Feld mit der Kennzeichnung 'ABBRUCH' anklicken.

Sie können sich den Inhalt der Datei aber auch auf einem an Ihren ST angeschlossenen Drucker ausgeben lassen (zu diesem Zweck das Feld mit der Kennzeichnung 'DRUCK' anklicken).

Wir wollen uns aber gemeinsam den Inhalt der Datei auf dem Bildschirm anschauen (zu diesem Zweck klicken Sie bitte das Feld mit der Kennzeichnung 'ANZEIGEN' an). Nun kann Unterschiedliches passieren; schaut Ihr Bildschirm vielleicht so aus:

```
rR␣H0000`x00000000

-ENDE DER DATEI-
```

oder so?

```
1ST WORD RELEASE NOTE

VERSION 1.02

This disk contains 1st Word,  the GEM word processor  written
by GST of Cambridge,  England and supplied with your Atari ST
computer.

In  order to provide you with a fully working word  processor
as  soon  as  possible,  the  1st Word User  Guide  has  been
supplied  on the microfloppy disk together with the  1st Word
software.

Backing up

Before you do anything else, backup the 1st Word disk.

Viewing the User Guide

-MEHR-█
```

Im erstgezeigten Fall handelt es sich ganz eindeutig um eine Datei, die entweder von einem Hauptprogramm (Blockdarstellung) während des Ladevorgangs automatisch von der Diskette direkt mit eingelesen wird (vielleicht ein Maschinenspracheprogramm) oder um ein vorher abgespeichertes Bild, das nur von einem Zeichenprogramm auf dem Monitorbildschirm eindeutig angezeigt werden kann. Geben wir den Inhalt solcher Dateien direkt aus - das haben Sie vielleicht gerade eben selbst erlebt - ist das Ergebnis nicht viel mehr als Kauderwelsch, den nur der Computer entziffern und somit gebrauchen kann.

Vielleicht wurde Ihnen aber auch Text auf dem Bildschirm ausgegeben, wie dies auf unserer zweiten Darstellung zu sehen ist. Vielleicht erscheint sogar in der untersten Bildschirmzeile die Bemerkung '--MEHR--', was soviel bedeutet wie: Bisher haben Sie nur einen Teil der Datei gesehen - mehr aber nicht. Drücken Sie hiernach die längliche Taste am unteren Rand der Tastatur (Leertaste), dann wird eine weitere Seite des Inhalts der Datei auf dem Bildschirm angezeigt. Drücken Sie allerdings die 'Enter'- oder die 'Return'-Taste, so wird lediglich eine weitere Zeile der angeklickten Datei auf dem Monitorbildschirm dargestellt. In diesem Fall zeigt Ihnen das 'lesbare' Monitorbild an, daß Sie wahrlich ein BASIC-Programm (Endabkürzung 'BAS'), oder aber einen Text von einem Textverarbeitungsprogramm (Endabkürzung meist 'DOC') aus dem Inhaltsverzeichnis herausgesucht und zur Anzeige auf dem Monitorbildschirm gebracht haben.

Sie haben hiermit die Möglichkeit, einmal kurz in Dateien hereinzuschauen. So können Sie z.B. feststellen, um was für ein BASIC-Programm es sich handelt, das sie nach Jahren irgendwo auf einer Ihrer Disketten finden, ohne gleich das BASIC zu laden und zu starten, das BASIC-Programm zu laden und zu listen. Sie sehen, hiermit ist eine große Zeitersparnis möglich.

Falls bei Ihnen auf dem Bildschirm bereits '--ENDE DER DATEI--' erscheint, bringt Sie das Andrücken einer beliebigen Taste zurück zum vertrauten Desktop oder Schreibtisch, von wo wir ja zuvor diese Darstellung gestartet haben. Ist ein Ende der Dateidarstellung auf dem Bildschirm noch nicht abzusehen (Sie werden durch die Anzeige '--MEHR--' in der letzten Bildschirmzeile immer von neuem dazu aufgefordert, die Leer- bzw. 'Enter'- bzw. 'Return'-Taste zu betätigen), können Sie den Anzeigevorgang auch dadurch abbrechen, daß Sie die Tasten 'Control' und 'C' gemeinsam oder die Taste 'Q' für Quit betätigen. Auf Ihrem Bildschirm müßte jetzt wieder das Inhaltsverzeichnis der in Ihrer Diskettenstation befindlichen Diskette erscheinen.

2.7 Die Tastatur

Sicher warten Sie schon darauf, endlich ein Programm auf Ihrem Rechner zu starten. Vorher wollen wir noch kurz sehen, wie die Tastatur des ST gebaut ist. Damit wir aber auch sehen, was passiert, wenn wir z.B. eine Buchstabentaste drücken, wollen wir ein Programm starten.

Sicher haben Sie schon einmal etwas über die Programmiersprache BASIC gehört. Sicher wissen Sie auch, daß auch Ihr ATARI ST diese Programmiersprache "spricht". Dies tut er aber nicht von Anfang an. Wir müssen ihm sozusagen die Vokabeln erst in seinen Speicher einladen.

Dies wollen wir nun tun (Legen Sie die zweite, ebenfalls im Lieferumfang Ihres ATARI ST enthaltene "LANGUAGE DISK" (= Sprachendiskette) in Ihr Diskettenlaufwerk, holen sich das Inhaltsverzeichnis (Doppelklick auf das Diskettenstationssymbol) und starten mit Doppelklick das Programm "BASIC.PRG".

Nach kurzer Zeit verändert sich der Bildschirm, und Sie erhalten ein Bild mit vier Fenstern. Lassen Sie das Aussehen zunächst noch unbeachtet, weil wir zunächst nur die Tastatur Ihres ST ausprobieren wollen.

Da Ihr ATARI ST in vielerlei Fällen ohne die Tastatur funktioniert (das ist ja gerade der Sinn der Maussteuerung mitsamt den zur Verfügung stehenden Entscheidungsfeldern und Pull-Down-Menüs), sind die meisten Tasten auf Ihrem Computer nicht bei jedem Programm in Funktion.

Zwar hören Sie auch in dem Fall, wenn gar kein Zeichen auf dem Bildschirm angezeigt wird (da keine Tastatureingabe möglich ist) ein Klicken bzw. sogar ein wiederholtes Klicken nach jedem Tastenanschlag, aber diese Eingaben per Tastatur sind dann wegen ihrer Nicht-Darstellbarkeit funktionslos.

Was aber passiert mit den einzelnen Tasten, wenn eine Bildschirmausgabe möglich ist (z.B. wenn Sie ein Textverarbeitungsprogramm in Ihren ST eingeladen haben)?

Auch dies läßt sich nicht pauschal festlegen, denn ATARI hat den Programmentwicklern in vielerlei Weise die Freiheit gelassen, die unterschiedlichen Tasten so zu belegen, wie es für das gerade bearbeitete Programm sinnvoll erscheint.

Wir können somit nicht auf jede Taste Ihres ATARI ST so eingehen, als wenn die Funktion immer gleich wäre, aber in vielerlei Fällen ist dies doch möglich.

Wir haben also die Programmiersprache BASIC eingeladen und können nun auch mit der Tastatur Eingaben machen. Was Sie eingeben, erscheint jetzt im COMMAND-Fenster (Befehlssfenster) links unten.

Die Tastatur Ihres ST besteht aus vier Blöcken:

1) Schreibmaschine
2) Funktionstasten
3) Editiertasten
4) Zehnerblock (wie bei einem Taschenrechner)

2.7.1 Die Schreibmaschine

Hier stehen Ihnen sämtliche Klein- und Großbuchstaben des Alphabets und vielerlei Sonderzeichen zur Verfügung. Während Sie die Großbuchstaben durch gleichzeitiges Andrücken von einer der beiden 'Shift'-Tasten auf den Bildschirm bringen können, müssen Sie bei Kleinbuchstaben nur die entsprechende Buchstabentaste drücken.

2.7.1.1 Alternate

Auf manchen Tasten wie z.B. der mit dem Umlaut 'Ä' finden Sie noch weitere Ausgabemöglichkeiten (hier sind es zusätzlich eckige bzw. geschweifte Klammern). Die untere Funktion (in diesem Fall die eckige Klammer) geben Sie durch gleichzeitiges Andrücken der Taste mit der 'Alternate'-Taste auf dem Bild-

schirm aus, während die obere Funktion (in diesem Fall die geschweifte Klammer) durch gleichzeitiges Andrücken der Buchstabentaste mit 'Shift' und 'Alternate' auf dem Bildschirm erscheint.

Probieren Sie das Andrücken von 'Alternate' bzw. von 'Alternate' und 'Shift' bitte noch mit anderen Tasten aus; in manchen Programmen werden Sie dadurch besondere Zeichenausgaben ermöglichen können.

2.7.1.2 Control

Schließlich besitzen manche Tasten Ihrer Schreibmaschinentastatur noch eine fünfte Ausgabemöglichkeit, indem Sie diese gleichzeitig mit der 'Control'-Taste betätigen. Bei einigen Programmen können Sie etwa durch die Tastenkombination 'Control'-'C' bzw. 'Control'-'G' eine Programmunterbrechung herbeiführen (z.B. bei der Programmiersprache BASIC). Sehen Sie in diesem Zusammenhang auch einmal in den Anhang A.

2.7.1.3 Shift

Wenn Sie eine andere Taste als die Buchstaben mit 'Shift' gemeinsam drücken, so gilt die Ausgabefunktion, die sich in der oberen Hälfte der Abbildung auf der entsprechenden Taste befindet (z.B. 'Shift'-'2' ergibt das Anführungszeichen oben '"').

2.7.1.4 Caps Lock

Die rechts von der Leerzeichen-Taste (auch 'Space' genannt) befindliche Taste mit der Aufschrift 'Caps Lock' ist ein An-/Ausschalter, der bei einmaligem Niederdrücken dafür sorgt, daß Sie ohne eine der 'Shift'-Tasten gleichzeitig drücken zu müssen, mit den Buchstabentasten Großschrift auf dem Bildschirm ausgeben können.

Drücken Sie die 'Caps Lock'-Taste ein zweites Mal, so wird der Urzustand wiederhergestellt (mit 'Shift' = Großbuchstaben, ohne 'Shift' = Kleinbuchstaben).

2.7.1.5 Tab

Die auf der linken Tastaturhälfte befindliche Taste mit der Aufschrift 'Tab' bedeutet eigentlich "Tabulator" und bewirkt die Fortbewegung des aktuellen Cursors auf dem Bildschirm um mehrere Zeichen nach rechts (der Cursor ist die Schreibposition auf dem Bildschirm, die meist durch einen senkrechten Strich oder einen schwarzen Block kenntlich gemacht ist).

Dies ist z.B. dann sinnvoll, wenn Sie Tabellen erstellen und die ausgegebenen Zahlen direkt untereinander stehen sollen.

2.7.1.6 Esc

Die 'Esc'-Taste (escape = ausbrechen) ist in vielen Programmen so etwas wie eine Rückschrittfunktion. So werden Sie z.B. in manchen Textverarbeitungsprogrammen nach Drücken der 'Esc'-Taste wieder rückwärts durch die einzelnen Menüs zum Ausgangspunkt zurückgeführt.

In Spielprogrammen dient die 'Esc'-Taste häufig zur Spielunterbrechung.

Beim Gebrauch des GEM-Betriebssystems, das wir in diesem Kapitel bereits ausführlich behandelt haben, dient die 'Esc'-Taste zu einem weiteren Zweck: Wenn Sie durch Doppelklicken das Inhaltsverzeichnis einer Diskette auf dem Bildschirm anzeigen, können Sie sich nach dem Wechsel der in Ihrer Diskettenstation befindlichen Diskette durch Drücken der 'Esc'-Taste das nun aktuelle Inaltsverzeichnis anzeigen lassen (Ihr ST zeigt Ihnen also wieder das aktuelle Inhaltsverzeichnis der zur Zeit in Ihrer Diskettenstation befindlichen Diskette an).

Noch eine weitere Funktion hat die 'Esc'-Taste bei Benutzung des GEM-Betriebssystems: Wenn Sie sich beispielsweise unter 'Info- Anzeige' den Namen einer Datei auf dem Bildschirm ausgeben lassen, können Sie diesen Namen durch Drücken der 'Esc'-Taste löschen, falls Sie einen anderen Namen eingeben wollen.

Schließlich verfügt Ihre ST-Schreibmaschinentastatur noch über drei weitere Sondertasten, die wir uns jetzt zusammen vorknöpfen wollen.

2.7.1.7 Backspace

Mit 'Backspace' (Backspace = Raum zurück, also Schritt zurück) löschen Sie das Zeichen, das links von Ihrem aktuellen Cursor steht. Wenn Sie die Taste festhalten oder hintereinander drücken, löschen Sie praktisch den Text rückwärts.

2.7.1.8 Delete

Mit 'Delete' (delete = streichen, löschen) löschen Sie das Zeichen, auf dem sich Ihr aktueller Cursor z.Zt. befindet. Wie sich allerdings inzwischen bei einigen Programmen ganz deutlich zeigt, haben die 'Delete'- und die 'Backspace'-Taste häufig die gleiche Funktion: Also vorsichtig, wie wir vorhin bereits betonten, die hier gemachten Aussagen zu verallgemeinern. Es ist für einen Programmierer ohne weiteres möglich, sämtliche Tasten der Tastatur z.B. mit dem griechischen Zeichensatz umzubelegen. So ist es auch kein Wunder, daß manchmal die Tasten 'Backspace' und 'Delete' die gleiche Funktion ausführen.

2.7.1.9 Return

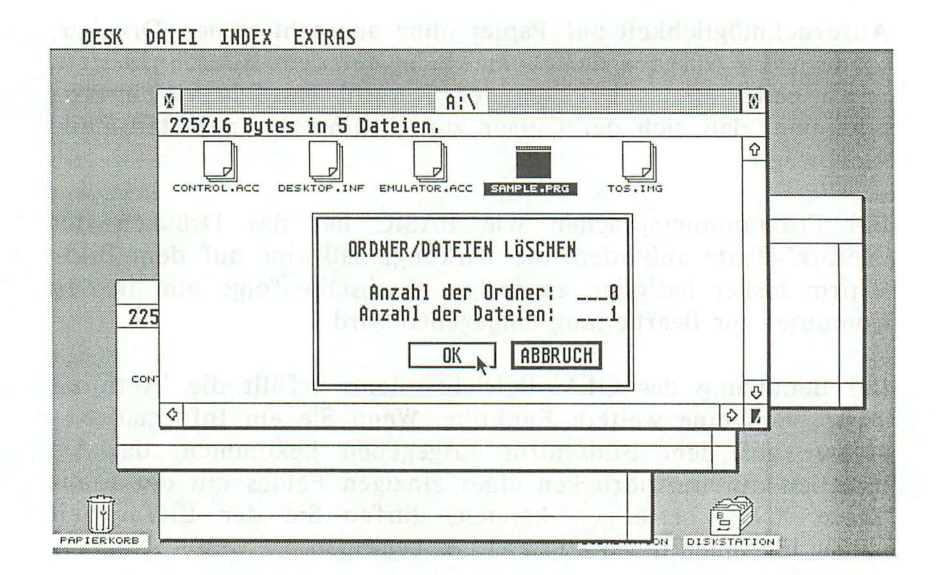
Schließlich gibt es noch die 'Return'-Taste (return = zurückkehren). Der Begriff 'Return' stammt von den Schreibmaschinen her, wo es meist eine Taste gibt, die den Wagenrücklauf, den 'Return', in die Wege leitet. Zwar besitzt Ihr ST keine direkte

Ausdruckmöglichkeit auf Papier ohne angeschlossenen Drucker, sondern die Ausgabe der Daten findet auf dem Bildschirm statt. Somit bedeutet das Drücken der 'Return'-Taste in vielen Programmen, daß sich der Cursor zum Beginn der nächsten Zeile hinbewegt.

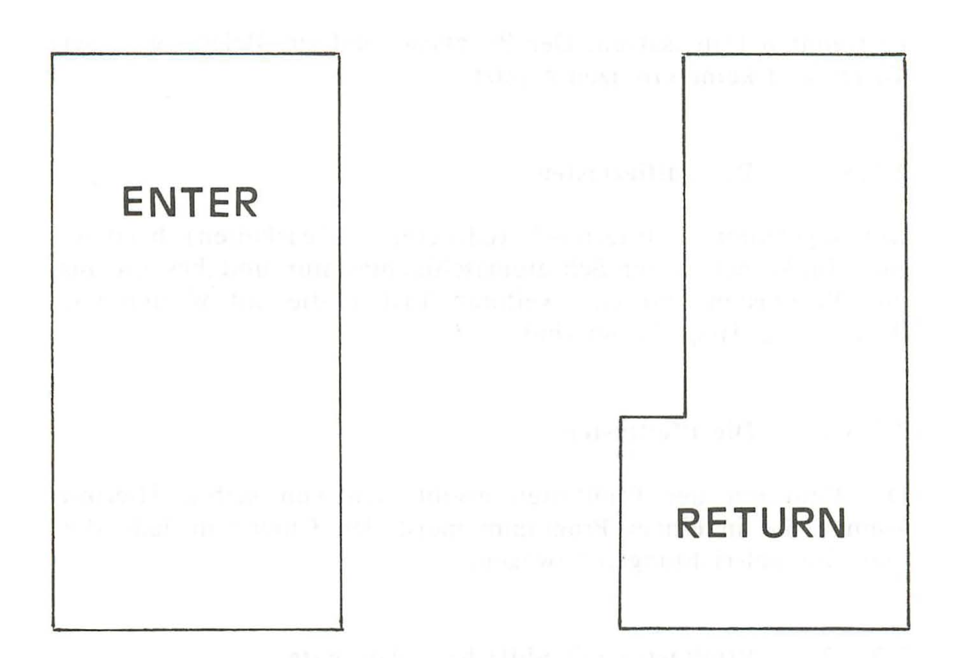
Bei Programmiersprachen wie BASIC hat das Drücken der 'Return'-Taste außerdem die Wirkung, daß eine auf dem Bildschirm bisher lediglich angezeigte Buchstabenfolge nun in den Computer zur Bearbeitung eingegeben wird.

Bei Benutzung des GEM-Betriebssystems erfüllt die 'Return'-Taste noch eine weitere Funktion: Wenn Sie ein Informationsfenster auf dem Bildschirm ausgegeben bekommen, das Sie lediglich durch Andrücken eines einzigen Feldes mit der Maus (meist 'OK') bestätigen können, dürfen Sie der Einfachheit halber hier auch die 'Return'-Taste betätigen.

Weiterhin bietet Ihnen Ihr ST unter GEM häufig die Möglichkeit, zwischen mehreren Feldern in einem Fenster auszuwählen. Ist eine dieser Auswahlmöglichkeiten durch einen fetter gemalten Kasten umgeben, bedeutet dies, daß Sie durch Drücken der 'Return'-Taste genau diese Funktion anwählen können.



Übrigens: Die 'Enter'-Taste im Zehnerblock hat im allgemeinen die gleiche Funktion wie die 'Return'-Taste. Wir nennen beide Tasten in diesem Buch Eingabetaste.



2.7.2 Die Funktionstasten

Zu den Funktionstasten F1 bis F10, die sich oberhalb der Schreibmaschinentastatur befinden, kann man leider keine allgemeinen Aussagen machen. Je nach Programm sind diese Tasten mit unterschiedlichen Funktionen belegt, wie z.B. bei einem Spiel vielleicht zur Einstellung des Schwierigkeitsgrades, bei einer Textverarbeitung zum Laden bzw. Speichern von Texten oder bei einer Datenbank zum Suchen und Finden von

bestimmten Datensätzen. Der Phantasie sind zur Belegung dieser Tasten fast keine Grenzen gesetzt.

2.7.3 Die Editiertasten

Der sogenannte Editierblock (editieren = bearbeiten) befindet sich direkt neben der Schreibmaschinentastatur und besteht aus vier Pfeiltasten und vier weiteren Tasten, die mit Worten wie 'Insert' und 'Help' belegt sind.

2.7.3.1 Die Pfeiltasten

Die Funktion der Pfeiltasten ergibt sich von selbst: Hiermit können Sie in einem Programm meist den Cursor in jede der vier Himmelsrichtungen bewegen.

2.7.3.2 Pfeiltasten mit Shift bzw. Alternate

Wollen Sie den Pfeil, den die Maus auf Ihrem Bildschirm anzeigt, mit den Pfeiltasten bewegen, müssen Sie gleichzeitig die 'Alternate'-Taste oder die 'Alternate'- und die 'Shift'-Taste betätigen. Im erstgenannten Fall bewegt sich der Mauspfeil sprunghaft (um jeweils 8 Punkte) in die gewünschte Richtung, während dies beim gleichzeitigen Drücken mit 'Shift' und 'Alternate' Punkt für Punkt gelingt.

2.7.3.3 Insert

Die 'Insert'-Taste dient in vielerlei Programmen dazu, Einfügungen in bereits bestehende Textstellen vorzunehmen. Dabei kommt der 'Insert'-Taste meist die Funktion eines An- /Ausschalters zu: Wird die 'Insert'-Taste einmal betätigt, so wird beim weiteren Schreiben der vom Cursor rechts befindliche Text nach rechts weiter verschoben, während ein erneutes Niederdrücken der 'Insert'-Taste wieder den Überschreibmodus einschaltet.

2.7.3.4 Clr/Home

Die 'Clr/Home'-Taste hat in einigen Programmen eine doppelte Funktion: Wird diese Taste gedrückt, kehrt der Cursor in die linke obere Bildschirmecke (nach Hause = 'Home') zurück; wird diese Taste hingegen gleichzeitig mit 'Shift' betätigt, passiert das gleiche wie ohne 'Shift', allerdings wird vorher zusätzlich der Bildschirm gelöscht (clear = löschen).

2.7.3.5 Help

Die 'Help'-Taste (= Hilfe) tut in vielen Programmen genau das, was ihr Name bereits aussagt: Weiß man nicht mehr weiter, drückt man die 'Help'-Taste und bekommt daraufhin meist eine Hilfestellung.

2.7.3.6 Undo

Wie ihr Name schon sagt, wird durch Drücken dieser Taste der zuletzt ausgeführte Vorgang wieder rückgängig gemacht ('Undo' = ungeschehen machen, aufheben).

2.7.3.7 Alternate mit Help gemeinsam

Drücken Sie 'Alternate' und 'Help' zusammen, so führt Ihr ST eine Ausgabe des aktuellen Bildschirms auf dem angeschlossenen Drucker aus. Sie sollten dies jedoch nur tun, wenn Sie wirklich einen Drucker betriebsbereit an Ihrem ST angeschlossen haben. Sollte dies nicht der Fall sein, so müssen Sie etwa 30 Sekunden warten, bis der Computer merkt, daß kein Drucker vorhanden ist und die Aktion abbricht. In dieser Zeit können Sie keine Eingaben machen, und auch der Mauspfeil gehorcht nicht den Mausbewegungen. Sollten Sie Schwierigkeiten mit dem Ausdruck

haben, so schauen Sie bitte im Anhang bei 'Druckeranpassung' nach. Dort wird beschrieben, wie Sie mit diesem Menüpunkt unter 'DESK' den Drucker richtig anpassen.

Wenn Sie die soeben genannten beiden Tasten während eines Ausdrucks ein zweites Mal betätigen, wird die Druckausgabe abgebrochen.

2.7.4 Der Zehnerblock

2.7.4.1 Enter

Schließlich noch ein paar Sätze zum Zehnerblock rechts außen. Über die 'Enter'-Taste sprachen wir bereits. Sie hat die gleiche Funktion wie 'Return'.

2.7.4.2 Die Ziffern, die Rechenzeichen und der Dezimal punkt

Die 10 Ziffern mitsamt den Rechen- und Sonderzeichen befinden sich ebenfalls auf der Schreibmaschinentastatur, nur ist hier in einem Block noch einmal alles vereint, was man bequem zusammenliegend gebrauchen kann.

Noch ein Hinweis: Da Ihr ST wie jeder andere Computer auch das Komma anders verwendet, als wir dies im deutschen Sprachraum gewohnt sind, muß man zur Abtrennung zwischen ganzen Zahlen und ihrem Dezimalanteil einen Punkt setzen und kein Komma. (Es heißt in der Computersprache also z.B. "1.50" und nicht "1,50".)

3. BASIC des ATARI ST

Im folgenden Kapitel wollen wir Ihnen eine Einführung in die Möglichkeiten von BASIC geben. Natürlich konnten wir nur grob umreißen, was man mit BASIC alles machen kann. Gleich zu Anfang wollen wir zusammen einige Befehle kennenlernen, einige kleine Programme selbst schreiben, und danach unser Wissen anhand von fertigen Programmen, die Sie selbst abtippen können, vertiefen. Haben Sie dann Spaß an der Programmierung des ATARI ST in dieser komfortablen Sprache gefunden, sollten Sie sich in anderen Büchern weiter informieren (z.B.: Das große BASIC-Buch zum ATARI ST, DATA BECKER 1985).

3.1 Wie lernt der ST BASIC?

3.1.1 BASIC - was ist das überhaupt?

Bisher waren wir eigentlich immer nur diejenigen, die dem ST auf irgendeine Frage oder einen Hinweis antworten mußten. Nun wird es höchste Zeit, daß eine richtige Unterhaltung zwischen uns und dem Computer in Gang kommt. Wir wollen uns mit ihm verständigen, möchten, daß er bestimmte Arbeiten für uns erledigt. Um dies zu erreichen, müssen wir allerdings seine Sprache erlernen. Gottseidank ist der ST so intelligent, daß er eine Vielzahl von Sprachen spricht. Diese Sprachen sind natürlich keine Fremdsprachen wie Französisch oder Spanisch, sondern Programmiersprachen.

Der ST ist, wenn er zu uns ins Haus kommt, leider aber nicht in der Lage, eine Programmiersprache zu verstehen. Alles was er dazu braucht, findet man aber auf der LANGUAGE DISK (Sprachendiskette), die dem ST beiliegt. Diese Diskette enthält also praktisch die Vokabeln und auch die Grammatik der entsprechenden Programmiersprache.

Wenn wir all das in seinen Speicher gebracht haben, können wir uns mit ihm über die Tastatur verständigen.

3.1.2 Einladen des BASIC

Jetzt endlich genug mit der grauen Theorie und hinein ins Vergnügen! Schließen Sie erst einmal alle offenen Fenster und legen Sie die Sicherheitskopie der Language-Diskette in Laufwerk A. Lassen Sie sich das Inhaltsverzeichnis anzeigen, und starten Sie das Programm "BASIC.PRG".

Jetzt kann es endlich losgehen.

3.2 Fensterln in BASIC

3.2.1 Funktion der Fenster in BASIC

Deutlich kann man es sehen: der Start des BASIC hat unseren Bildschirm ganz schön verändert. Die Menüleiste in der ersten Zeile trägt nicht nur andere Bezeichnungen als wir es bisher gewohnt waren, sondern auch sämtliche vier möglichen Bildschirmfenster wurden für BASIC gleich mitbenutzt:

1) Der OUTPUT-Bildschirm, auf dem die Programmausgabe stattfindet.

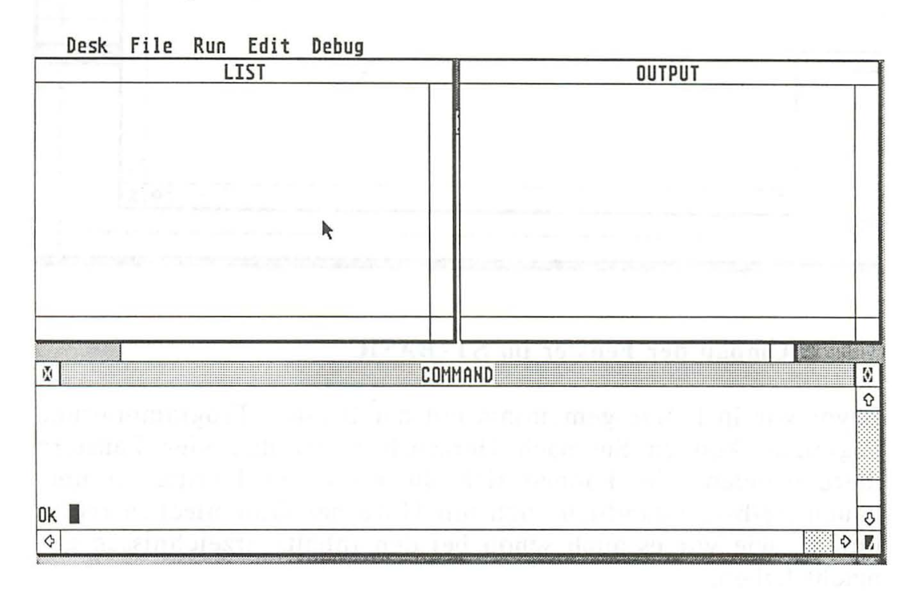
2) Der COMMAND-Bildschirm (command = Befehl), auf dem Sie die Programmeingaben vornehmen.

3) Der LIST-Bildschirm, der für die Darstellung von Programmlistings (Auflistung des Programmes) gedacht ist.

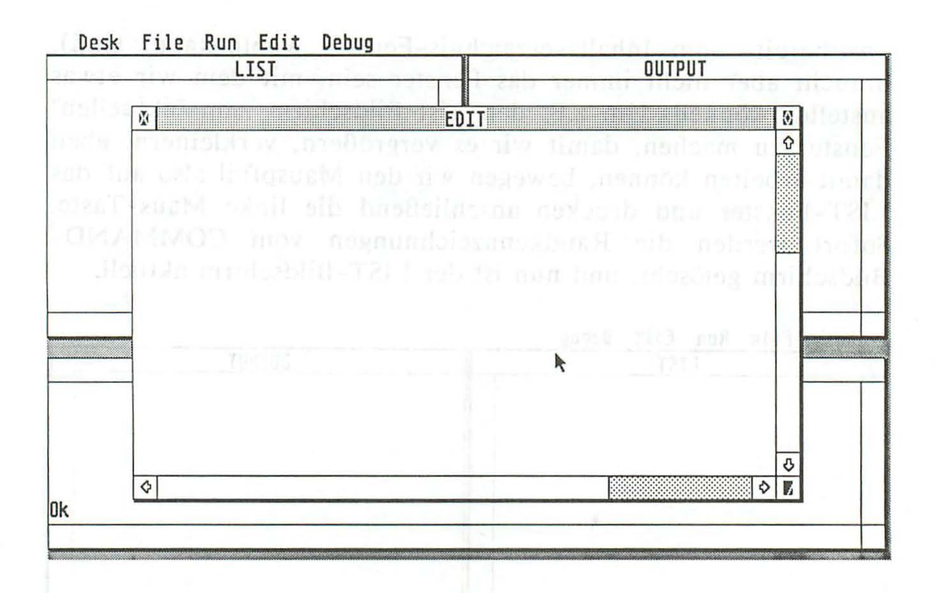
4) Der EDIT-Bildschirm (dieser Bildschirm ist nur teilweise durch die drei anderen Bildschirme sichtbar), auf dem Sie später Programmverbesserungen vornehmen können.

Der zum Programmbeginn vom ST angezeigte COMMAND-Bildschirm (die Ansteuerung dieses z.Zt. aktuellen Fensters erkennen Sie an den vorhandenen Randkennzeichnungen, wie sie

uns bereits vom Inhaltsverzeichnis-Fenster wohlbekannt sind), braucht aber nicht immer das Fenster sein, mit dem wir etwas anstellen können. Um z.B. den List-Bildschirm zum "aktuellen" Fenster zu machen, damit wir es vergrößern, verkleinern, eben damit arbeiten können, bewegen wir den Mauspfeil also auf das LIST-Fenster und drücken anschließend die linke Maus-Taste. Sofort werden die Randkennzeichnungen vom COMMAND-Bildschirm gelöscht, und nun ist der LIST-Bildschirm aktuell.



Den EDIT-Bildschirm sehen wir zwischen den drei anderen Bildschirmen an den Randstellen in der Mitte durchschimmern. Bewegen Sie den Mauspfeil auf diese kleine Stelle, klicken einmal mit der linken Maustaste, und augenblicklich wird Ihnen nun auch der EDIT-Bildschirm als aktueller Bildschirm nach vorne geholt.



3.2.2 Umbau der Fenster im ST-BASIC

Bevor wir in Kürze gemeinsam mit der BASIC- Programmierung beginnen, können Sie nach Herzenslust mit den vier Fenstern 'herumspielen'. Sie können sich die einzelnen Fenster so umbauen (selbstverständlich auch mit Hilfe der Gummiecken rechts unten), wie wir es auch schon bei den Inhaltsverzeichnissen gemacht haben.

3.3 Erste Kontakte mit BASIC

Zuerst wollen wir darauf hinweisen, daß es zwei verschiedene BASIC-Versionen für Ihren ST gibt. Die beiden Versionen können Sie anhand der Programmlänge unterscheiden. Die ältere Version hat eine Programmlänge von 138944 Bytes, die neue hat nur noch ein Länge von 123988 Bytes. Sie können die Versionen aber auch anhand des Menüpunktes "Desk" unterscheiden. Bei der älteren Version erscheint "About ST Basic", bei der neueren "About MCC Basic". Die von uns beschriebenen Programme laufen auf beiden BASIC-Versionen, jedoch unterscheiden sich

die eventuell auftretenen Fehlermeldungen im genauen Wortlaut voneinander. Wir werden bei den entsprechenden Stellen im Buch auf beide Varianten eingehen. Eine komplette Auflistung der BASIC-Fehlermeldungen finden Sie im Anhang.

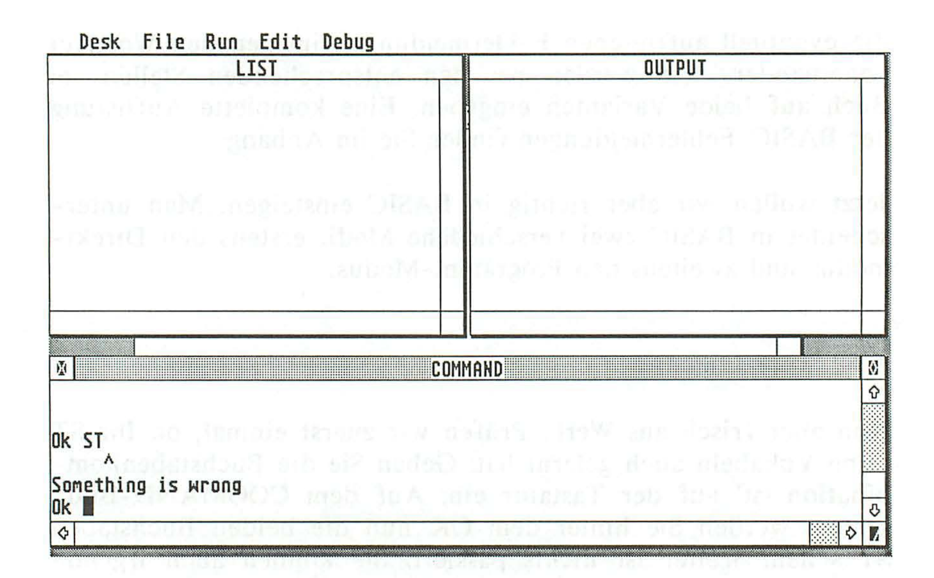
Jetzt wollen wir aber richtig in BASIC einsteigen. Man unterscheidet in BASIC zwei verschiedene Modi, erstens den Direktmodus und zweitens den Programm-Modus.

3.3.1 Direktmodus

Nun aber frisch ans Werk. Prüfen wir zuerst einmal, ob Ihr ST seine Vokabeln auch gelernt hat: Geben Sie die Buchstabenkombination 'st' auf der Tastatur ein. Auf dem COOMAND-Bildschirm werden Sie hinter dem OK nun die beiden Buchstaben ST sehen. Weiter ist nichts passiert. Sie können auch irgendwelche anderen Buchstaben- oder Zeichenkombinationen in Ihren ST eingeben; das einzige, was hier passiert, ist die gleichzeitige Bildschirmdarstellung, die Sie direkt Zeichen für Zeichen am Bildschirm sehen können.

Warum passiert denn nun nichts? Der ST wartet praktisch noch, was wir mit ihm tun wollen. Um ihm nun klarzumachen, daß wir auf unser liebevoll eingetipptes "ST" eine Reaktion von ihm erwarten, müssen wir es seinem Speicher "übergeben". Dies geschieht mit der RETURN oder ENTER-Taste, die wir ja "Eingabetaste" nennen wollten.

Drücken wir einfach einmal die RETURN-Taste. Der ST zeigt sofort Reaktion, 'Something is wrong' bzw. 'Syntax Error'im neuen BASIC (was soviel bedeutet wie: Irgendwas ist falsch in der gewählten Vokabel - die kenne ich nicht) ist da zu lesen.



Obwohl unser Computer die Bezeichnung ST trägt, ist er nicht in der Lage, etwas mit dieser Vokabel anzufangen - ST gehört nicht zu den ca. 150 Ihrem Computer bekannten Vokabeln.

Tippen Sie ruhig ein paar andere englische Wörter auf gut Glück, vielleicht auch Ihren Namen oder den von anderen Personen, in den Computer ein. Fast sind wir sicher, daß Sie außer der obigen Meldung nichts anderes aus ihm herauskitzeln werden, nachdem Sie die Eingabetaste betätigt haben.

'Deprimierend, wie klein der BASIC-Wortschatz meines Rechners doch eigentlich ist!' werden Sie vielleicht nun sagen. Zudem besitzt Ihr Computer nach der eintönigen Beantwortung Ihrer Eingabe auch jedesmal noch die Frechheit, die für ihn abschließende Bemerkung 'Ok' auf dem Bildschirm auszugeben.

Dazu gleich ein Hinweis, um diese Verurteilung augenblicklich zu entschärfen: Es gibt Zeitpunkte, wo Sie zwar eine Eingabe auf der Tastatur vornehmen können, die entsprechende Ausgabe aber gar nicht auf dem Bildschrim erscheint. In diesem Fall

würde auch keine 'Ok'-Antwort erfolgen. Mit "OK" will uns der ST nur sagen, daß er wieder bereit ist, Befehle von uns entgegenzunehmen.

Um nun endlich auch einmal etwas anderes von unserem ATARI ST zu sehen, geben Sie doch einmal folgenden Befehl ein:

```
PRINT "ATARI ST"
```

Anschließend betätigen Sie die Eingabetaste. Was passiert? Vorausgesetzt, Sie haben alles richtig abgetippt, steht auf dem OUTPUT-Bildschirm nun "ATARI ST". Fein, gell?

Bewirkt hat dies das Wort "PRINT", das zu dem Wortschatz Ihres ST gehört. Innerhalb der Anführungszeichen können Sie nun statt "ATARI ST" auch den Namen Ihres Partners, Ihres Rauhaardackels oder was auch immer eingeben. Stets wird der ST brav Ihren Befehl ausführen, nachdem Sie die Eingabetaste gedrückt haben.

Man kann also dem ST direkt Befehle geben, etwas zu tun, solange man dies mit einem Wort macht, das er in seinem Wortschatz hat. Diese Art der Eingabe nennt man *Direktmodus*.

3.3.2 Programmodus

Um im Direktmodus den ATARI ST etwas machen zu lassen, muß man die ganze Zeile immer wieder neu eingeben. Was aber, wenn man will, daß die gleiche Arbeit immer wieder ausgeführt wird? Man schreibt ein *Programm*.

Um ein solches Programm zu schreiben, wurde schließlich die Programmiersprache BASIC erfunden. Hat man einmal ein Programm geschrieben und eingetippt, kann man es so oft ausführen lassen, wie man will und kann den Computer durch einen einzigen Befehl dazu veranlassen, das Programm zu starten. Diese Art der Arbeit nennt man *Programmodus*.

3.4 Der BASIC-Grundwortschatz

Wir wollen nun lernen, wie man ein solches Programm schreiben kann, damit wir den ST dazu bringen können, für uns zu arbeiten. Dazu wollen wir einige der wichtigsten BASIC-Befehle und Techniken kennenlernen. Löschen wir zuerst einmal den OUTPUT-Bildschirm mit "CLEARW 2". Diesen Befehl tippen Sie einfach ein. Er bewirkt das Löschen des 2. Fensters (CLEAR = löschen; W = Abkürzung für WINDOW = Fenster; 2 = Nummer des Fensters). Nun löschen wir den COMMAND-Bildschirm mit "CLEARW 3".

3.4.1 Die Zeilennummern

Wenn man seinen ST in BASIC programmiert, muß man jeder Zeile eine Nummer geben. Nach dem "OK" fängt man also nicht gleich mit dem Befehl an, sondern stellt erst einmal eine Nummer voran. Zweckmäßig hat sich als erste Zeilennummer die Zahl "10" erwiesen. Anschließend kann nun der Befehl folgen. Man fährt dann bei der zweiten Zeile am besten mit "20" fort usw. Dieser 10er-Abstand hat den Vorteil, daß man später leicht auch noch Zeilen einfügen kann, die dann z.B. "15" oder "17" heißen.

3.4.2 Ein- und Ausgabe: INPUT und PRINT

Zu den wichtigsten Befehlen in BASIC gehören die Ein- und Ausgabebefehle. Ohne sie könnte man weder Werte eingeben, noch würde man ein Ergebnis auf dem Bildschirm zu sehen bekommen.

3.4.3 Der PRINT-Befehl

Wie wir vorher schon gesehen haben, dient PRINT dazu, etwas auf dem Bildschirm auszugeben. Mit der Zeile

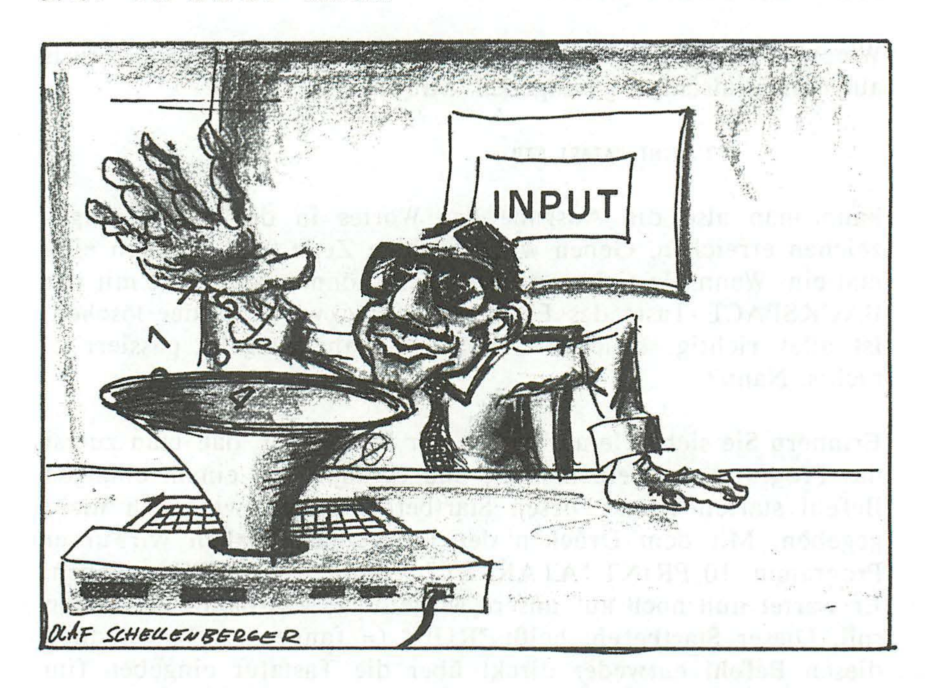
```
10 PRINT "ATARI ST"
```

kann man also die Ausgabe des Wortes in den Anführungszeichen erreichen. Geben wir die obige Zeile doch einfach einmal ein. Wenn Sie sich vertippt haben, können Sie leicht mit der BACKSPACE-Taste das Eingegebene rückwärts wieder löschen. Ist alles richtig, drücken Sie die Eingabetaste. Es passiert ... nichts. Nanu?

Erinnern Sie sich? Gerade haben wir festgestellt, daß man zuerst das Programm schreiben muß, und es dann mit einem einzigen Befehl starten kann. Diesen Startbefehl haben wir noch nicht gegeben. Mit dem Drücken der Eingabetaste haben wir unser Programm '10 PRINT "ATARI ST"' lediglich dem ST übergeben. Er wartet nun noch auf unsere Mitteilung, was damit geschehen soll. Dieser Startbefehl heißt "RUN" (= fange an). Wir können diesen Befehl entweder direkt über die Tastatur eingeben (im Direktmodus, also *ohne* Zeilennummer), oder wir fahren mit der Maus auf das Pull-Down-Menü "RUN" und klicken dort das Wort "RUN" an. Auf dem COMMAND-Bildschirm erscheint nun ebenfalls "RUN", und auf dem OUTPUT-Bildschirm sehen wir das Resultat unseres Programmes, nämlich "ATARI ST".

Schön, werden Sie sagen, das gleiche Ergebnis haben wir aber auch bekommen, als wir die Zeile ohne Zeilennummer eingetippt haben. Richtig! Aber gleich werden Sie den Unterschied deutlich sehen.

3.4.4 Der INPUT- Befehl



Zuerst löschen wir wieder, wie oben, mit "clearw 2" und "clearw 3" die OUTPUT- und COMMAND-Fenster. Nun ist unser erstes einzeiliges Programm nicht mehr zu sehen. Trotzdem ist es aber noch im Speicher des ST enthalten. Deshalb löschen wir nun den Speicher mit "NEW", das wir einfach, ohne Zeilennummer, eingeben.

Oben haben wir immer den Text, den der ST ausgeben sollte, zwischen den Anführungszeichen eingetippt. Nun werden wir aber langsam bequemer und wollen deshalb ein etwas komfortableres Programm schreiben. Wir wollen erreichen, daß uns der ST fragt, welchen Text er ausgeben soll, und nach unserer Antwort dann unseren Wunsch ausführt. Dazu benutzen wir den INPUT-Befehl.

Schreiben wir also zunächst:

```
10 INPUT "WAS SOLL ICH SCHREIBEN";A$
20 PRINT A$
```

Nun klicken wir wieder RUN an - und was passiert? Im OUTPUT-Fenster erscheint die Frage unseres ST "Was soll ich schreiben?" Nun können wir irgend etwas eingeben. Danach drücken wir die Eingabetaste, und auf dem OUTPUT-Fenster erscheint unser Text noch einmal.

Wie haben wir das gemacht? In der Zeile 10 unseres Programmes passiert folgendes:

Nach dem INPUT-Befehl kann man einen Kommentartext in Anführungsstrichen eingeben, der dann vom Computer als Fragetext benutzt wird. Das Fragezeichen wird vom ST selbst gesetzt. Wenn man nun auf seine Frage hin etwas eingetippt hat, die Eingabetaste gedrückt hat, dann gibt der ST diesem Text den Namen "A$". Diesen Namen nennt man "Variable". Das Dollarzeichen bedeutet, daß es sich bei dem Eingegebenen um einen Text handelt. Wollen wir nur Zahlen eingeben, mit denen vielleicht später im Programm gerechnet werden soll, müssen wir das Dollarzeichen weglassen.

In Zeile 20 gibt der ST nun den Text mit dem Namen "A$" aus. Dann ist er fertig mit dem Programm, und im COMMAND-Fenster erscheint wieder das "OK".

Also:

Der INPUT-Befehl ordnet einer Variablen einen Text oder eine Zahl zu.

Der PRINT-Befehl bewirkt die Ausgabe eines Textes oder des Inhaltes einer Variablen.

3.4.5 Unbedingtes Springen: GOTO



Eines der einfachsten und geradezu sinnlosesten Programme, die doch immer wieder von Computerfreaks voller Stolz auf den Rechnern eingegeben werden, besteht aus nur zwei Zeilen. Damit Sie diesen Freaks nun auch etwas zu entgegnen haben, geben Sie folgenden Zweizeiler auch in Ihren ST ein (vorher löschen Sie bitte den Speicher mit NEW und die 3 Bildschirme mit CLEARW 1, CLEARW 2 und CLEARW 3):

```
10 PRINT "ATARI ST"
20 GOTO 10
```

Starten Sie das Programm wieder mit "RUN". Was passiert? Im OUTPUT-Fenster erscheint immer wieder "ATARI ST". Wenn Sie nun aufhören würden, in diesem Buch zu lesen, könnten Sie

das Programm nicht mehr anhalten und müßten somit Ihren ST vom Stromnetz trennen bzw. zumindest den Reset-Schalter betätigen.

Doch dies ist eine sehr unelegante Möglichkeit, das Programm zu unterbrechen. Viel einfacher gelingt dies, indem Sie die Control-Taste mit der Taste G gleichzeitig drücken. Je nachdem, wo sich das Programm gerade befindet, wird nun auf dem Bildschirm angezeigt: '-- Break -- at line 10' bzw. '-- Break -- at line 20' (Unterbrechnung des Programmablaufs in Zeile 10 bzw. in Zeile 20).

Wir könnten das Programm anschließend auch von der gleichen Stelle fortfahren lassen, indem wir nun den Befehl CONT (CONT ist die Abkürzung für Continue, was soviel bedeutet wie: weitermachen) eingeben.

Was besagen die zwei Zeilen?

Zeile 10: Mit PRINT wird das Zauberwort "ATARI ST" auf dem Bildschirm ausgegeben.

Zeile 20: "GOTO 10" bewirkt, daß wieder zur Zeile 10 gesprungen wird, und wieder wird "ATARI ST" ausgegeben usw., bis wir das Programm abbrechen.

Nun gibt es beim PRINT-Befehl aber auch noch einige Kleinigkeiten zu erwähnen, die die Darstellung auf dem Bildschirm verändern. Tippen Sie bitte noch einmal die Zeile 10 des Programmes ein und hängen Sie ein Komma an, bevor Sie die Eingabetaste drücken. Die Zeile sollte jetzt so aussehen:

```
10 PRINT "ATARI ST",
```

Starten Sie das Programm mit RUN und jetzt sollten drei Kolonnen mit "ATARI ST" in Ihrem Output-Fenster sichtbar sein. Auch hier brechen Sie das Programm mit CTRL-G ab. Was bewirkt nun das angehängte Komma? Nun, das Komma besitzt beim PRINT-Befehl eine Art Tabulator-Funktion, die Sie vielleicht schon von einer Schreibmaschine her kennen. Beim

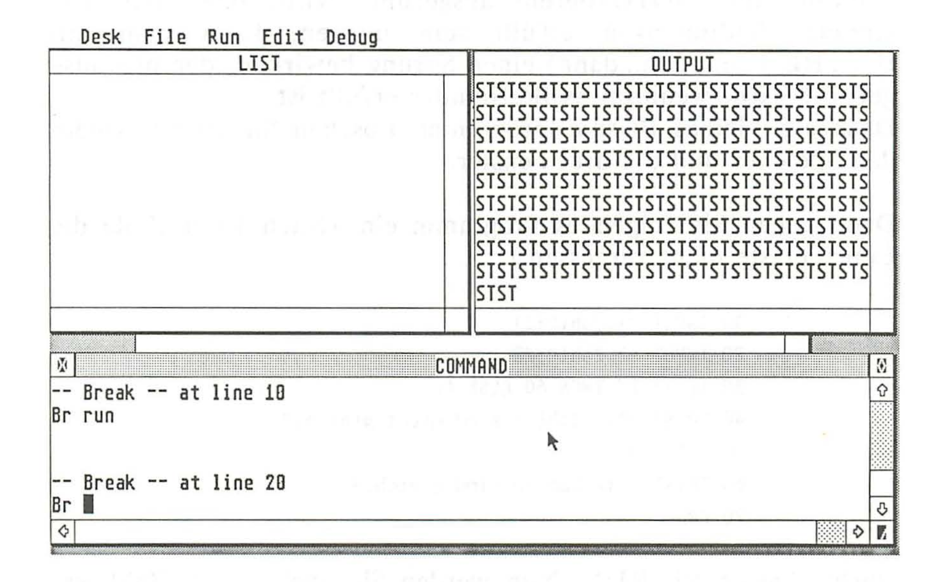
BASIC ist der Abstand der Tabulatoren auf 15 gesetzt, das heißt, daß nach einem Komma an der nächsten freien 15er Position weitergeschrieben wird. Verwechseln Sie das nicht mit dem Abstand, denn Sie sehen an unserem Beispiel, daß der Abstand kleiner als 15 Zeichen beträgt und abhängig von der Länge der geschriebenen Worte ist. Mit dieser Tabulator-Funktion können Sie z.B. kleinere Tabellen ordentlich untereinander auf den Bildschirm bringen.

Ein weiteres kleines Anhängsel ist das Semikolon. Löschen Sie bitte das Programm, das wir noch im Speicher haben, mit "NEW" und Eingabetaste, und tippen Sie folgendes kurze Programm ein.

```
10 PRINT "ST";
20 GOTO 10
```

Sollten Sie sich bei einer Zeile vertippt haben, so können Sie, solange noch nicht die Eingabetaste gedrückt wurde, mit der Taste 'BACKSPACE' die Zeile rückwärts löschen, genau wie beim INPUT-Befehl. Ist die Eingabetaste jedoch einmal gedrückt, hilft im Moment nur neueintippen der Zeile.

Starten Sie das Programm mit RUN und betätigen der Eingabetaste. Jetzt sollten Sie etwa folgendes Bild haben.



Sie sehen, daß ein Semikolon hinter dem PRINT-Befehl bewirkt, daß alles hintereinander geschrieben wird.

Der Befehl, den wir in diesem Abschnitt aber eigentlich kennenlernen wollten, GOTO, dürfte durch unser Miniprogramm klargeworden sein.

Also:

Der GOTO-Befehl bewirkt das Springen zu jeder beliebigen Zeilennummer, die man hinter GOTO setzt.

3.4.6 Wenn...dann...ansonsten...: IF...THEN...ELSE...

Während der GOTO-Befehl ausgeführt wird, ohne daß bestimmte Bedingungen erfüllt sein müssen, kann man mit IF...THEN (= wenn...dann) einen Sprung bewirken, der nur ausgeführt wird, wenn etwas Bestimmtes erfüllt ist.
Dazu ein kleines Beispielsprogramm. Löschen Sie zuerst wieder den Speicher und die drei Fenster.

Dann geben Sie folgendes Programm ein. (Nach jeder Zeile die Eingabetaste nicht vergessen!)

```
10 INPUT "1.Zahl";Z1
20 INPUT "2.Zahl";Z2
30 IF Z1=Z2 THEN 60 ELSE 40
40 PRINT "Die Zahlen sind nicht gleich."
50 GOTO 70
60 PRINT "Die Zahlen sind gleich."
70 END
```

Starten Sie es mit RUN. Nun werden Sie nach der 1. Zahl gefragt. Geben Sie irgendeine Zahl ein und drücken die Eingabetaste. Dann folgt die Frage nach einer zweiten Zahl. Nach der Eingabe der zweiten Zahl und Drücken der Eingabetaste steht da plötzlich noch ein weiterer Satz. Ihr ST hat uns ausgegeben, ob die beiden Zahlen gleich sind, oder nicht.
Wie funktioniert das nun?

Die Zeilen 10 und 20 sind uns verständlich. Die erste eingegebene Zahl nennen wir "z1" (ohne Dollarzeichen, weil eine Zahl) und die zweite "z2".
Zeile 30 untersucht nun, ob z1 gleich z2 ist.
Zwei Möglichkeiten gibt es nun:

1. Die Zahlen sind gleich
Dann geht der ST zur Zeile 60, und es wird ausgegeben, daß die Zahlen gleich sind. Nun springt der ST nicht wieder zurück, sondern macht an dieser Stelle weiter und geht nun zur nächsten Zeile, wo er aufgefordert wird, zur Zeile 70 zu gehen, wo das Programm beendet ist.

2. Die Zahlen sind nicht gleich
Dann geht der ST zur Zeile 40. Warum?
Der Befehl IF...THEN...ELSE... bewirkt, daß *wenn* etwas erfüllt ist, *dann* zur angegebenen Zeile gesprungen wird. Wenn die Bedingung *nicht erfüllt* ist, dann geht der Computer zur Zeilennummer, die hinter ELSE steht. In diesem Falle also: *wenn* z1=z2 ist, *dann* gehe zur Zeile 60, *ansonsten* gehe zur Zeile 40.
Der ST ist also bei Zeile 40 angelangt, wo er ausgibt, daß die Zahlen nicht gleich sind. Anschließend geht er zur nächsten Zeile, wo er aufgefordert wird, zur Zeile 70 zu gehen, wo das Programm beendet ist.

Also:

Mit IF...THEN...ELSE kann man Sprünge bewirken, die an bestimmte Bedingungen geknüpft sind.

3.4.7 Schleifen mit FOR...NEXT

Es wäre doch fein, wenn wir bestimmen könnten, wie oft der ST das Programm durchlaufen soll, also wie oft er nach dem Ende wieder von vorne anfangen soll, bevor er ganz mit dem Programm aufhört. Um dies zu erreichen, benutzt man eine sogenannte "Schleife". Lassen wir unser Programm im Speicher (also *nicht* "NEW" tippen!!) löschen aber die drei Fenster.
Jetzt geben wir noch folgende vier Zeilen ein.

```
5 INPUT "Wieviele Durchgänge";D
6 FOR I=1 TO D
70 NEXT I
80 END
```

Wollen Sie wissen, wie unser Programm jetzt aussieht? Nichts einfacher als das! Haben Sie nach jeder Zeile immer die Eingabetaste gedrückt? Fein, dann können wir uns jetzt das gesamte Programm auflisten lassen. Fahren Sie dazu in das Pull-Down-

Menü EDIT und klicken den Begriff LIST an. Augenblicklich tut sich endlich auch im LIST-Fenster einmal etwas. Das gesamte Programm steht nun da:

```
5 INPUT "Wieviele Durchgänge";D
6 FOR I=1 TO D
10 INPUT "1.Zahl";Z1
20 INPUT "2.Zahl";Z2
30 IF Z1=Z2 THEN 60 ELSE 40
40 PRINT "Die Zahlen sind nicht gleich."
50 GOTO 70
60 PRINT "Die Zahlen sind gleich."
70 NEXT I
80 END
```

Was haben wir gemacht? Wir haben die Zeilen 5 und 6 dem Programm hinzugefügt. Die Zeile 70 hieß vorher einfach:

```
70 END
```

Dadurch, daß wir eine neue Zeile ebenfalls mit der Nummer 70 bedacht haben, hat der ST die alte Zeile 70 gelöscht und die neue Zeile 70 in seinen Speicher aufgenommen. Die Zeile 70 heißt jetzt:

```
70 NEXT I
```

Die Zeile 80 haben wir ebenfalls angefügt. Haben Sie gesehen, wie einfach es ist, seine Programme zu erweitern? Man braucht nur die neuen Anweisungen einzugeben, das Alte muß nicht mehr mit abgetippt werden, weil es noch im Speicher enthalten ist.

Was haben wir nun verändert? Die Zeile 5 ist sicherlich klar. Die Variable d bekommt den Wert der Anzahl der von Ihnen gewünschten Programmdurchläufe. Dann folgt Zeile 6. Hier ist der sogenannte Schleifenanfang. Das Schleifenende finden wir in Zeile 70. Alles was dazwischen liegt ist also innerhalb der Schleife.

Folgen wir einfach einmal dem Programm.

Zeile 5: Anzahl der Durchgänge in Schublade d.
Zeile 6: Schleifenanfang. Die Variable i (auch "Zähler" genannt) wird um den Wert 1 erhöht. Dies kann man nicht aus dem Befehl sehen, weil der FOR...NEXT-Befehl sehr viel leistet. Dann wird geprüft, ob i schon gleich d ist. Solange dies noch nicht der Fall ist, wird nun die folgende Schleife ausgeführt.
Zeilen 10 bis 60: Programm wie oben.
Zeile 70: Sprung zum Schleifenanfang (Zeile 6). Dort wird dann i wieder um 1 erhöht und geprüft, ob i schon gleich d ist. Wenn nein, wird die Schleife wieder ausgeführt usw. bis i gleich d ist. Dann Sprung zur Zeile 80 (Programmende).

Wir haben gesehen, FOR...NEXT ist ein sehr leistungsfähiger Befehl, der manches tut, was man ihm nicht ansieht.

3.4.8 Unterprogramme: GOSUB...RETURN

Im Gegensatz zum GOTO-Befehl hat der GOSUB-Befehl den Vorteil, daß Unterprogramme nicht nur von einer Stelle, sondern von beliebig vielen Stellen angesprungen werden können. Nach Ablauf des Unterprogramms macht der Computer an der Stelle weiter, von wo das Unterprogramm aufgerufen worden ist. Um Ihnen den genauen Verlauf zu demonstrieren, hier wieder ein kleines Beispielprogramm:

```
5 CLEARW 2
10 GOSUB 100
20 PRINT "Sie haben eingegeben: ";
30 PRINT A$
40 GOSUB 100
50 PRINT "Jetzt haben Sie eingegeben: ";
60 PRINT A$
70 GOSUB 100
80 PRINT "Und nun: ";
90 PRINT A$
```

```
95 END
100 PRINT
110 INPUT "Bitte geben Sie ein Wort ein ";A$
120 RETURN
```

Nun zur Funktionsbeschreibung:

Zeile 5: Das Output-Fenster wird gelöscht.

Zeile 10: Das Programm verzweigt in das Unterprogramm in Zeile 100.

Zeile 100: Das PRINT erzeugt eine Leerzeile.

Zeile 110: Hier wird der Text "Bitte geben Sie ein Wort ein" ausgegeben und auf eine Eingabe gewartet. Das eingegebene Wort steht dann in A$.

Zeile 120: Das RETURN bewirkt, daß das Unterprogramm verlassen wird. Der Programmablauf setzt hinter dem Unterprogrammaufruf ein, hier also in Zeile 20.

Zeile 20: Hier wird der Text "Sie haben eingegeben: " ausgegeben.

Zeile 30: Da in Zeile 20 ein Semikolon nach dem PRINT-Befehl stand, wird hier das Wort aus A$ direkt hinter dem vorherigen Satz ausgegeben.

Zeile 40: Hier wird wieder das Unterprogramm aufgerufen. Nach Ablauf des Unterprogramms wird der Programmablauf bei Zeile 50 fortgesetzt.

Zeile 50: Nun wird der Text "Jetzt haben Sie eingegeben " ausgegeben. Durch das anschließende Semikolon wird der nachfolgende Text direkt im Anschluß ausgegeben.

Zeile 60: Jetzt wird Ihr eingegebenes Wort ausgegeben.

Zeile 70: Wieder wird in das Unterprogramm in Zeile 100 gesprungen. Nachdem Sie das neue Wort eingegeben haben, wird das Programm in Zeile 80 fortgeführt.

Zeile 80: Jetzt wird der Text "Und nun: " ausgegeben.

Zeile 90: Im direkten Anschluß erscheint nun Ihr drittes eingegebenes Wort.

Zeile 95: Hiermit endet das Programm.

Sie sehen, daß wir den Programmteil zur Eingabe des Wortes nicht dreimal, sondern nur einmal schreiben brauchten. Der Computer merkt sich, von wo das Unterprogramm aus aufgerufen worden ist und führt nach Beendigung des Unterprogramms mit RETURN den Programmablauf mit dem auf den Aufruf folgendem Befehl fort. So kann man oft wiederkehrende Unterprogramme einmal schreiben und sie von mehreren Stellen aufrufen. Beim GOTO-Befehl weiß der Computer nicht, von wo er aufgerufen worden ist. GOTO ist also nur für Sprünge ohne Rückkehr geeignet.

3.4.9 Noch bessere Übersicht: REM

Wenn Sie an den Beginn einer Zeile die Buchstabenkombination REM schreiben (Abkürzung für REMARK = erinnern), übersieht Ihr ST diese Zeile und fährt mit dem Programm eine Zeile tiefer fort.

Für uns haben REM-Zeilen allerdings eine sehr wichtige Bedeutung: Wir können uns hinter der Buchstabenfolge REM Dinge notieren, die für die Lesbarkeit unserer Programme sehr hilfreich sind.

Verändern wir unser GOSUB-Beispielprogramm von vorhin mit Hilfe der REM-Zeilen:

```
10 REM Programm zur GOSUB-Demonstration
20 REM Sprung ins Bildschirm-Ausgabe-Unterprogramm
30 GOSUB 100
40 REM Programmneustart
50 GOTO 10
100 REM Bildschirm-Ausgabe-Unterprogramm
110 PRINT "ATARI ST"
120 RETURN
```

Vor dem Programmstart mit RUN sollten Sie sich das Listing mit LIST anschauen.

```
Desk  File  Run  Edit  Debug

LIST
  10    REM Programm zur GOSUB-Demon
  20    REM Sprung ins Bildschirm-Au
  30    GOSUB 100
  40    REM Programmstart
  50    GOTO 10
  100   REM Bildschirm-Ausgabe-Unter
  110   PRINT "ATARI ST"
  120   RETURN

OUTPUT
ATARI ST
ATARI ST
ATARI ST
ATARI ST
ATARI ST
ATARI ST
ATARI ST
ATARI ST
ATARI ST

COMMAND
Ok 100 REM Bildschirm-Ausgabe-Unterprogramm
Ok 110 PRINT "ATARI ST"
Ok 120 RETURN
Ok list
Ok run
```

Selbst wenn Sie dieses Programm noch nach Jahren lesen, werden Sie schnell verstehen, was hier geschehen ist.

Also:

Die REM-Zeilen helfen uns dabei, unsere Programme übersichtlicher und damit besser verständlich zu machen.

3.4.10 Fenster putzen: CLEARW



Bestimmt erinnern Sie sich noch daran, wie wir vor dem Start eines Programmes jedesmal die Fenster gelöscht haben. Dafür gibt es den Befehl

```
CLEARW N
```

Der Buchstabe "n" kann nun Werte von 0 bis 3 annehmen, also vier verschiedene, genau so viele, wie es Fenster im BASIC des ST gibt. Dabei bedeutet:

n=0: Löschen des EDIT-Fensters
n=1: Löschen des LIST-Fensters
n=2: Löschen des OUTPUT-Fensters
n=3: Löschen des COMMAND-Fensters

3.4.11 Als wäre nichts geschehn: NEW

Auch das ist für uns schon ein alter Hut. Will man alles im Speicher löschen, also Variablen und auch das Programm, so ist NEW der richtige Befehl. Aber Vorsicht! Erst überlegen, dann tippen!

3.5 Etwas über Variablen

Sie haben es sicher schon bemerkt: Ohne Variablen, diese Buchstaben, die ganze Sätze oder auch Zahlen "in sich" aufnehmen können, geht in der Computerei nichts. Fangen wir einmal bei den Zahlen an.

3.5.1 Einfache Zahlenvariablen

Zuerst einmal soll sich Ihr ATARI ST also Zahlen merken. Wir richten zu diesem Zweck eine Schublade in seinem Speicher ein und geben dieser einen Namen. Bisher haben wir unterschiedliche Namen genommen (a, A$, z1, usw.) Jetzt nehmen wir einfach einmal den Namen "st". Unsere Schublade soll also "st" heißen.

Bisher haben wir immer einer Variablen einen Wert (eine Zahl oder einen Text) zugeordnet, indem wir die INPUT-Anweisung benutzt haben. Es geht aber auch einfacher. Wie in der Mathematik, müssen wir dazu eine Gleichung aufstellen.

Wenn wir z.B. unserem ST zu verstehen geben wollen, daß die Schublade mit dem Namen ST den Zahlwert 6 enthält, schreiben wir einfach:

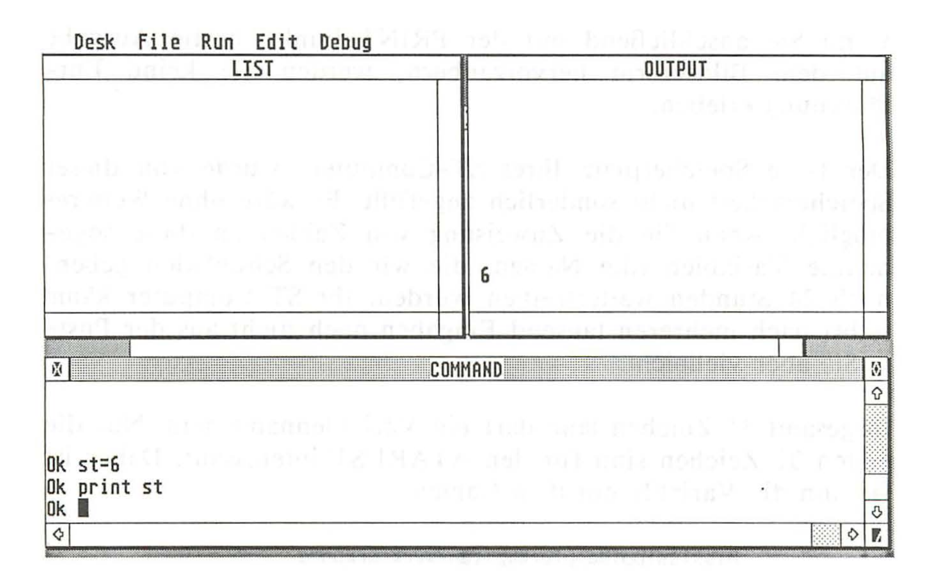
```
ST=6
```

Vergessen Sie nicht, wiederum anschließend die Eingabetaste zu betätigen. Fein. Keine Fehlermeldung wurde auf dem COMMAND- Bildschirm ausgegeben, sondern nun steht da ganz zufriedenstellend 'Ok'. Unser ST hat sich gemerkt, daß die Schublade ST den Wert 6 enthält!

Sie zweifeln? Probieren Sie es doch selbst einmal aus, indem Sie die Schublade ST auf den OUTPUT-Bildschirm ausgeben. Wie war das noch mit der Bildschirmausgabe? Richtig! PRINT heißt das Zauberwort. Um den Inhalt der Schublade mit dem Namen "st" auszugeben, tippen wir fröhlich:

```
PRINT ST
```

Haben Sie bis hierhin alle Punkte befolgt, sollte nun auf Ihrem OUTPUT-Bildschirm die Zahl 6 erscheinen.



Vielleicht haben Sie noch ein gesundes Mißtrauen, denn eine Zeile vor Ihrer Eingabe haben Sie ja noch selbst geschrieben: 'ST=6'. Wie leicht (?) könnte Ihr ST einfach dort abgeschrieben haben! Um auch diesen Beweis anzutreten (daß nicht gemogelt wurde, und die Schublade ST sich nicht nur auf dem Bild-

schirm, sondern viel tiefer in Ihrem ST-Computerspeicher befindet) löschen Sie den Bildschirm einfach durch Eingabe des Befehls: CLEARW 2 für den OUTPUT-Bildschirm und CLEARW 3 für den COMMAND-Bildschirm und geben erneut den Befehl zur Speicherabfrage ein:

```
PRINT ST
```

Wieder wurde der Zahlwert 6 ausgegeben. Für Sie als gesunden, zweifelnden Zeitgenossen hoffentlich ein schlüssiger Beweis! Geben Sie nun auch anderen Schubladen Namen und Zahlwert:

```
Agnes=4
Brigitte=5
Claudia=7
Doris=6
```

Wenn Sie anschließend mit der PRINT-Funktion die Ausgabe auf dem Bildschirm hervorzaubern, werden Sie keine Enttäuschung erleben.

Der freie Speicherplatz Ihres ST-Computers wurde von dieser Speicherarbeit nicht sonderlich angefüllt. Es wäre ohne Weiteres möglich, wenn Sie die Zuweisung von Zahlen an diese sogenannte Variablen (die Namen, die wir den Schubladen geben) noch 24 Stunden weitertreiben würden. Ihr ST-Computer käme selbst nach mehreren tausend Eingaben noch nicht aus der Puste ... Sie aber vielleicht.

Insgesamt 31 Zeichen lang darf ein Variablenname sein. Nur die ersten 31 Zeichen sind für den ATARI ST interessant. Daher ist für ihn die Variable mit dem Namen

```
Mississippidampferkapitänswerksurkunde
```

dieselbe Variable wie

```
Mississippidampferkapitänswerkspatent
```

weil eben nur die ersten 31 Zeichen (bis zum "s" von "...werks" vom ST beachtet werden. Sie werden aber zugeben, daß 31 Zeichen für einen Variablennamen in der Regel ausreichen.

Es gilt noch zwei weitere Punkte bei der Auswahl der Variablennamen zu beachten:

1. Der Variablenname darf kein reserviertes BASIC-Wort sein. So wäre es nicht möglich, eine Variable 'Print' zu nennen; probieren Sie es ruhig einmal praktisch aus:

```
PRINT=5
```

Ergebnis: `Something is wrong`

denn das ist die uns bereits vertraute ST-Vokabel PRINT.

2. Das erste Zeichen eines Variablennamens muß ein Buchstabe sein (Kleinbuchstabe oder Großbuchstabe ist dabei egal, denn hier unterscheidet Ihr ST nicht).

Von Ihrem ST anerkannte Variablennamen wären demnach z.B.: A3, Ac, B., aber natürlich auch X, Y und C.

Die Zuweisung würde bei folgenden Variablennamen jedoch nicht funktionieren: 4A und 1E.

3.5.2 Ganzzahl- und Dezimalzahlvariable

Wir wollen noch weitere Zahlen in unserem ST ablegen, jedoch werden wir jetzt darauf achtgeben, ob sich die hohe Stellengenauigkeit unseres Computers bei mathematischen Aufgaben bzw. Zahlen auch reduzieren läßt.

Was heißt das? Tippen Sie doch einfach einmal ein

```
PRINT 2/7
```

(dieser Strich bedeutet "geteilt durch")

Ergebnis: 0.285714

Mit insgesamt sechs Stellen hinter dem Komma hat unser ST diese Divisionsaufgabe gelöst.

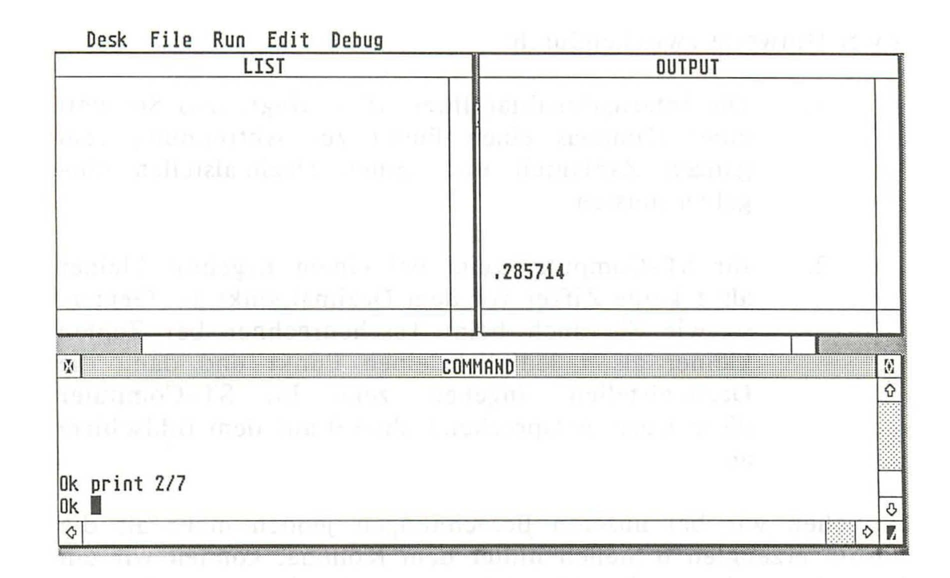
Wie können wir nun diese Berechnung in einer Variablen abspeichern? Wir suchen uns wieder einen Variablennamen (behalten wir ganz einfach den Namen ST bei) und weisen diesem entweder das Endergebnis Ziffer für Ziffer zu (in unserem Fall 'ST=0.285714') oder lassen das Ergebnis während des Eingabevorgangs direkt berechnen:

```
ST=2/7
```

Wenn Sie nun

```
PRINT ST
```

eingeben, so erscheint auf dem OUTPUT-Bildschirm das Ergebnis:



Sie können sich vielleicht vorstellen, daß die Abspeicherung einer Zahl mit so vielen Ziffern hinter dem Komma mehr als einen Speicherplatz verbraucht.

Wenn Sie nun aber kein Mathematiker sind bzw. nur das kleine Einmaleins mit Ihrem Computer berechnen wollen, wäre es schade, so viel Speicherplatz sinnlos zu verschenken (zudem rechnet Ihr Computer selbstverständlich erheblich schneller, wenn er keine Stellen hinter dem Komma darstellen soll). Reicht Ihnen eine verminderte Dezimalstellengenauigkeit der auszugebenden Zahl aus, so müssen Sie den Variablennamen dafür kenntlich machen: Sie setzen ein Prozentzeichen hinter den Variablennamen, wenn Ihnen null Stellen hinter dem Komma genügen.

```
ST%=2/7
PRINT ST%
```

Ergebnis: 0.

Zwei Hinweise zwischendurch:

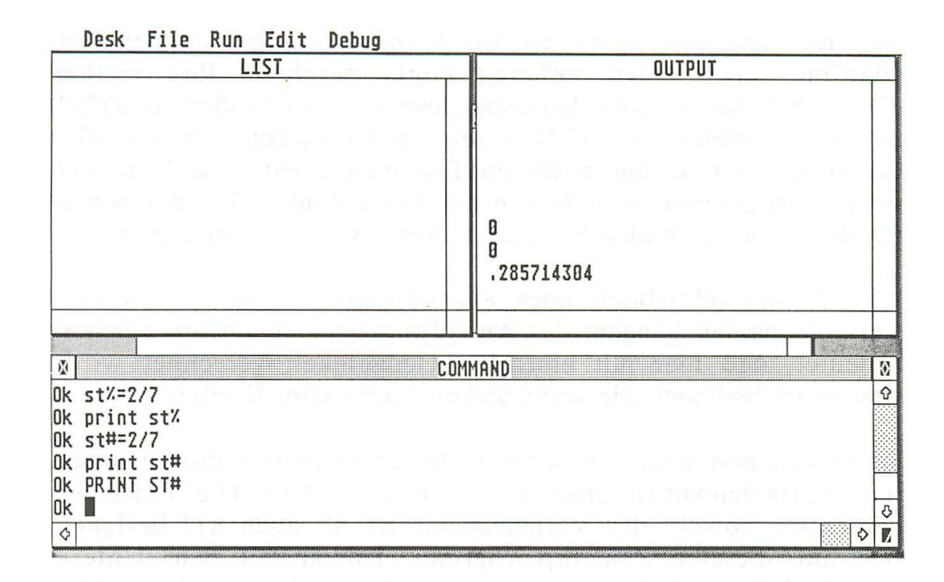
1. Die Internationalität Ihres ST bedingt, daß Sie statt eines Kommas einen Punkt zur Abtrennung vom ganzen Zahlanteil und seinen Dezimalstellen eingeben müssen.

2. Ihr ST-Computer zeigt bei einem Ergebnis kleiner als 1 keine Ziffer vor dem Dezimalpunkt an. Genauso wie Sie auch beim Taschenrechner bei Zahlen kleiner als 1 lediglich einen Punkt und dann die Dezimalstellen eingeben, zeigt Ihr ST-Computer diese Werte entsprechend ohne 0 auf dem Bildschirm an.

Brauchen wir bei unseren Berechnungen jedoch mehr als die vorhin erzeugten 6 Stellen hinter dem Komma, können wir ein Doppelkreuz hinter dem Variablennamen eingeben also, z.B.

```
ST#=2/7
PRINT ST#
```

Ergebnis: 0.285714304



Das Ergebnis hinter der sechsten Stelle kommt allerdings in dieser Rechenart nur annähernd an den richtigen Zahlwert heran.

Bedenken Sie deshalb bei Divisionsaufgaben immer, ob sich der Speicherplatzvorteil und die Rechengeschwindigkeit im Vergleich zur Rechengenauigkeit gut miteinander vertragen! Das Schöne ist, daß wir mit Hilfe der unterschiedlichen Genauigkeitsangaben gleich dreimal einen Variablennamen verwenden können. Geben Sie zur Kontrolle ein:

```
print st
```

Ergebnis: 0.285714

```
print st%
```

Ergebnis: 0

```
print st#
```

Ergebnis: 0.285714304

Bei der ganzzahligen Verarbeitung von Variablen müssen wir allerdings noch einen weiteren Punkt beachten: Ihre Zahlen dürfen bei dieser Variablenabspeicherung nicht größer als 32767 und nicht kleiner als -32767 sein, sonst merkt sich Ihr ST-Computer anstelle der positiven Eingaben negative Zahlen und umgekehrt positive anstelle von negativen Zahlen. Bei der neuen BASIC-Version können Sie jedoch Werte von $\pm 2^{31}$ eingeben.

Wir können schließlich noch ein weiteres Zeichen hinter den Variablennamen hängen: das Ausrufungszeichen. Dieses Zeichen bedeutet, daß hier mit einfacher Genauigkeit gerechnet wird, was soviel bedeutet wie sechs Stellen hinter dem Komma.

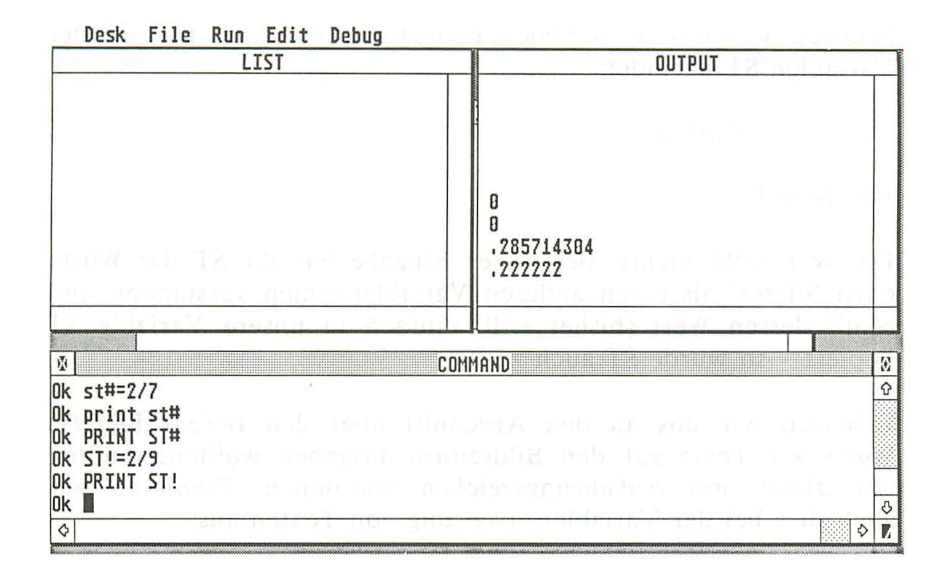
Sie fragen nun sicher, welcher Unterschied denn dann zu unserem Variablennamen ohne Zusatz besteht (ST). Die Frage ist berechtigt. Sowohl der Variabenname ST als auch ST! beziehen sich auf dieselbe Zahlabspeicherung, jedoch mit dem Unterschied, daß durch Eingabe des Ausrufungszeichens hinter dem Variablennamen die entsprechende Variable noch einmal für einfache Genauigkeit besonders kenntlich gemacht wird.

```
ST!=2/9
PRINT ST!
```

Ergebnis: 0.222222

```
PRINT ST
```

Ergebnis: 0.222222



Da die Variable ST! der Variablen ST entspricht, ändert sich somit auch der Inhalt der Variablen ST, sie nimmt den gleichen Wert an wie die Variable ST!.

Hiermit vorerst einmal genug zur Zahlenvariablenabspeicherung. Wie können wir nun auch noch Texte im Speicher des ST ablegen?

3.5.3 Text- oder Stringvariable

Vielleicht wundern Sie sich, daß wir die Abspeicherung von Texten in Variablen hier gesondert behandeln. Warum sollte es nicht möglich sein, Texte auch in unseren bisherigen Variablentypen abzulegen?

Probieren wir es einfach aus. Wir wollen das Wörtchen 'dieser' in der uns bereits bekannten Variablen ST abspeichern:

```
ST=DIESER
```

Schauen wir uns doch gleich einmal an, was sich nun in der Variablen ST befindet:

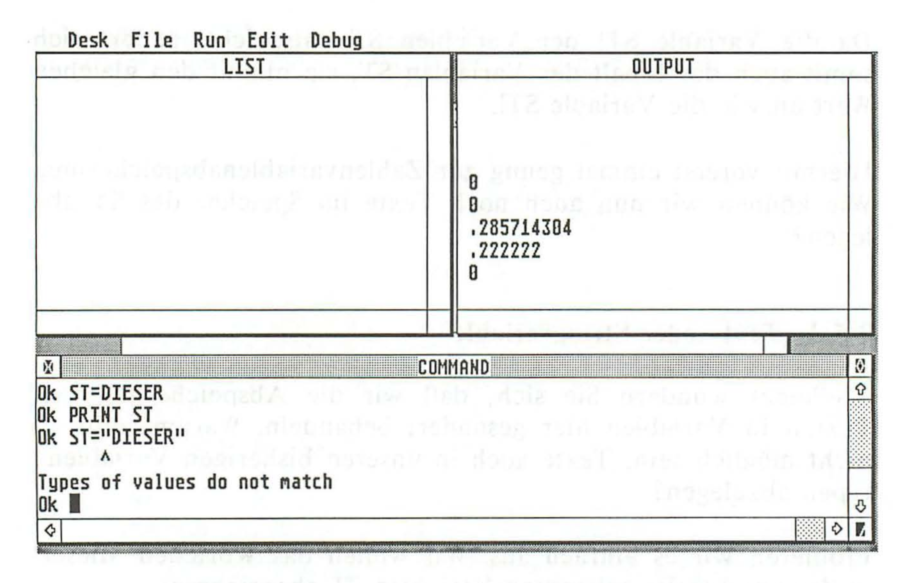
```
PRINT ST
```

Ergebnis: 0

Das war wohl nichts. Bei dieser Eingabe hat Ihr ST das Wörtchen 'dieser' als einen anderen Variablennamen verstanden und somit dessen Wert (bisher = 0) einfach in unsere Variable ST kopiert - so wurde ST auch zu 0.

Erinnern wir uns an den Abschnitt über den Befehl PRINT: Wenn wir Texte auf den Bildschirm ausgeben wollten, mußten wir diesen mit Anführungszeichen umrahmen. Probieren wir auch dies bei der Variablenzuweisung von Texten aus:

```
ST="DIESER"
```



Ergebnis: Eine neue Fehlermeldung namens 'Types of value do not match' (frei ins Deutsche übersetzt würde das bedeuten: Du hast den falschen Variablentyp verwendet, die Zahlvariable ST

ist zur Textabspeicherung nicht geeignet!). Bei der neuen BASIC-Version wird die Fehlermeldung 'TYPE MISMATCH' ausgegeben. Die Bedeutung ist jedcoh genau die gleiche.

Aus diesem Grund müssen wir noch ein anderes Anhängsel für die Kennzeichnung von Variablennamen dazulernen: das Dollarzeichen $.

Fügen wir dieses Zeichen einem Variablennamen hinzu, so entsteht eine sogenannte String- oder Zeichenkettenvariable, die dazu in der Lage ist, Texte von einer Länge bis zu 255 Zeichen zu speichern. Probieren wir dieses aus, indem wir unser Beispiel von vorhin mitsamt dem richtigen Variablentyp anwenden:

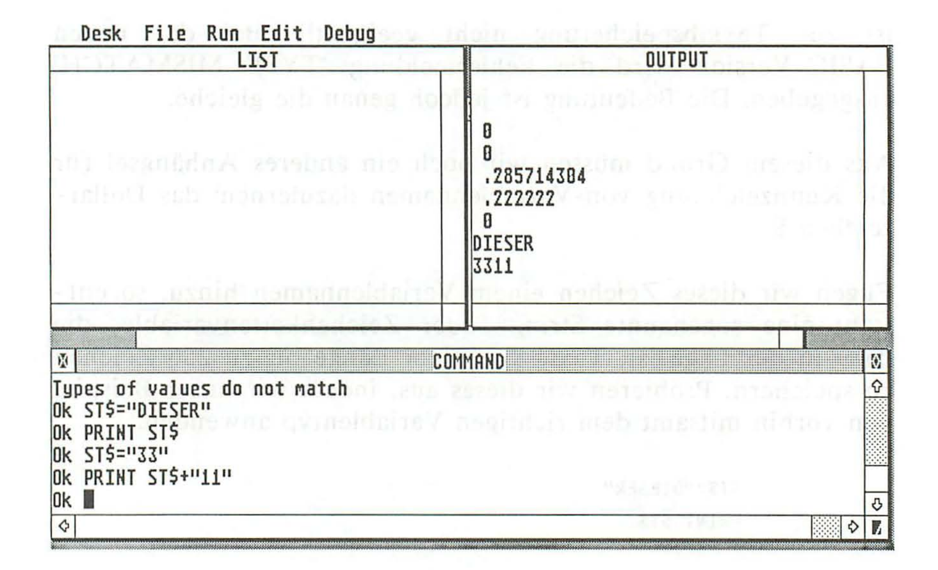
```
ST$="DIESER"
PRINT ST$
```

Ergebnis: dieser

Sie können selbstverständlich auch Zahlen in einer Stringvariablen abspeichern. Zu diesem Zweck müssen Sie allerdings auch diese in Anführungszeichen einschließen (sonst wird wieder die Fehlermeldung 'Types of value do not match' auf dem Bildschirm angezeigt) und zudem: Rechnen läßt sich mit einer Stringvariablen nicht direkt. Probieren Sie es aus:

```
ST$="33"
PRINT ST$+"11"
```

Ergebnis: 3311



Hier wurden nicht mehr die Zahlen rein mathematisch addiert, sondern die beiden Texte einfach aneinandergefügt: Die Stringvariable ST$ mit dem Text "33" und der eingegebene Textzahlenwert "11" verschmolzen zu 3311.

Vorhin haben wir festgestellt, daß Textvariablen 'nur' bis zu 255 Zeichen enthalten dürfen ("dieser" wären z.B. 6 Zeichen).

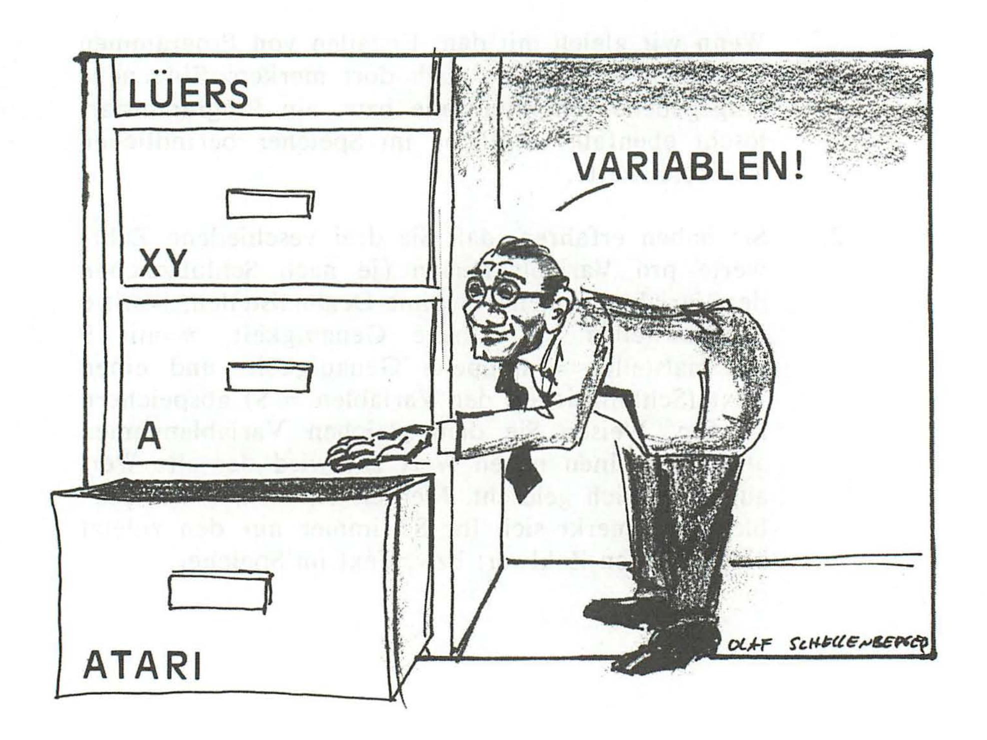
Sie können somit erkennen, daß z.B. eine DIN-A-4-Seite mit ca. 2000 Zeichen nicht in eine einzige Textvariable paßt, sondern mindestens 8 zur Speicherung benötigt.

Noch zwei Hinweise:

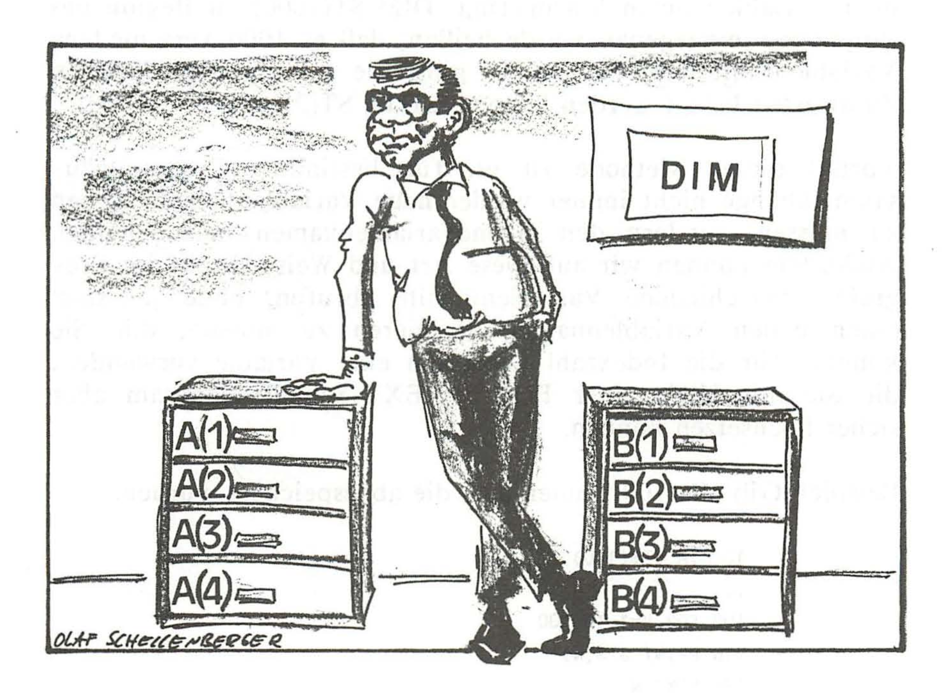
1. Wenn Sie den Befehl CLEAR eingeben, werden augenblicklich in Ihrem Computer alle variablen Zahlwerte und Texte gelöscht. Geben Sie deshalb am besten immer zu Beginn Ihrer Arbeit mit dem ST das CLEAR-Kommando ein.

Wenn wir gleich mit dem Erstellen von Programmen beginnen, werden Sie auch dort merken: Eine neue eingegebene Programmzeile bzw. ein Programmstart löscht ebenfalls sämtliche im Speicher befindlichen Variablenwerte.

2. Sie haben erfahren, daß Sie drei veschiedene Zahlwerte pro Variablennamen (je nach Schlußzeichen des Variablennamens: %=ohne Dezimalstellen, !=mit 6 Dezimalstellen = einfache Genauigkeit, #=mit 9 Dezimalstellen = doppelte Genauigkeit) und einen Text (Schlußzeichen der Variablen = $) abspeichern können. Weisen Sie dem gleichen Variablennamen allerdings einen neuen Wert zu, wird der alte Wert augenblicklich gelöscht. Merken Sie sich: Pro Variablenname merkt sich Ihr ST immer nur den zuletzt eingegebenen Zahlwert bzw. Text im Speicher.


LÜERS
XY
A
ATARI
VARIABLEN!
OLAF SCHELLENBERGER

3.5.4 Dimensionierte Variable



Sie erinnern sich: Gerade haben wir festgestellt, daß wir vier verschiedene Variablen mit dem gleichen Namen verwenden können (String $ für Texte, doppelte Genauigkeit mit #, einfache Genauigkeit mit ! und ganze Zahlen mit %), aber halt nicht mehr.

Mit dem Befehl DIM hingegen legen wir für eine Variable eine sogenannte Dimension (Ausbreitung) fest, so daß wir denselben Variablennamen auch 100- oder 1000mal mit unterschiedlichen Werten verwenden dürfen, aber halt immer mit einer Indexzahl versehen, die zusätzlich angegeben werden muß.

Diese Indexzahl wird nach der Dimensionierung in Klammern dem Variablennamen hinzugefügt. DIM ST(1000) zu Beginn des Programms eingegeben würde heißen, daß es 1000 verschiedene Variablen mit dem Namen ST gibt, die mit unterschiedlichen Zahlwerten belegt werden können: ST(1), ST(2) ... bis ST(1000).

Vorteil dieser Methode ist es, für bestimmte Programmzusammenhänge nicht immer wieder neue Variablennamen suchen zu müssen, sondern den Grundvariablennamen beizubehalten. Außerdem können wir auf diese Art und Weise direkt im Programm verschiedene Variableninhalte abrufen, ohne jedesmal einen neuen Variablennamen definieren zu müssen, d.h. Sie könnten für die Indexzahl wiederum eine Variable verwenden, die Sie mit Hilfe einer FOR ... NEXT-Schleife langsam aber sicher hochsetzen können.

Beispiel: Gib alle 100 Namen aus, die abgespeichert wurden:

```
10 DIM ST$(100)
...
100 FOR N=1 TO 100
110 PRINT ST$(N)
120 NEXT N
```

Vorausgesetzt, wir hätten 100 verschiedene Namen in der dimensionierten Variablen ST$ abgespeichert, würde dieser Vierzeiler zur Ausgabe vollkommen reichen.

Anders ohne die Möglichkeit der Dimensionierung: Wir müßten alle 100 Variablen einzeln im Programm, Zeile für Zeile, zum Ausdruck bringen:

```
10 PRINT ST$
20 PRINT ST1$
30 PRINT ST2$
...
```

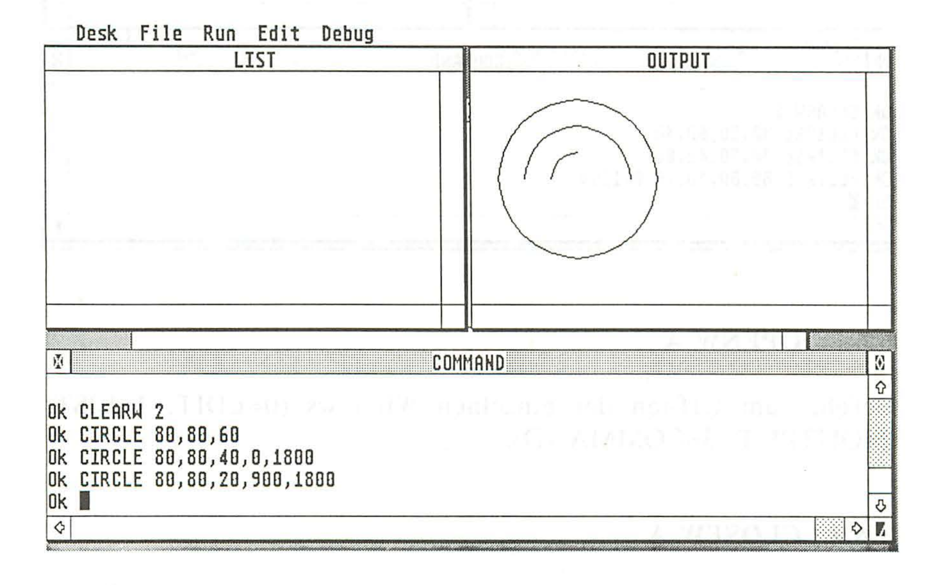
Eine Tortur ohnegleichen.

3.6 Interessante Grafikbefehle des ST-BASIC

Die wichtigsten BASIC-Befehle haben wir nun kennengelernt. Bevor wir jetzt zu größeren Programmen zum Abtippen kommen, noch ganz kurz einige interessante Grafikbefehle, mit denen Sie einfach einmal herumexperimentieren können. Anschließend ein paar nützliche Helfer, die das Programmieren leicht machen.

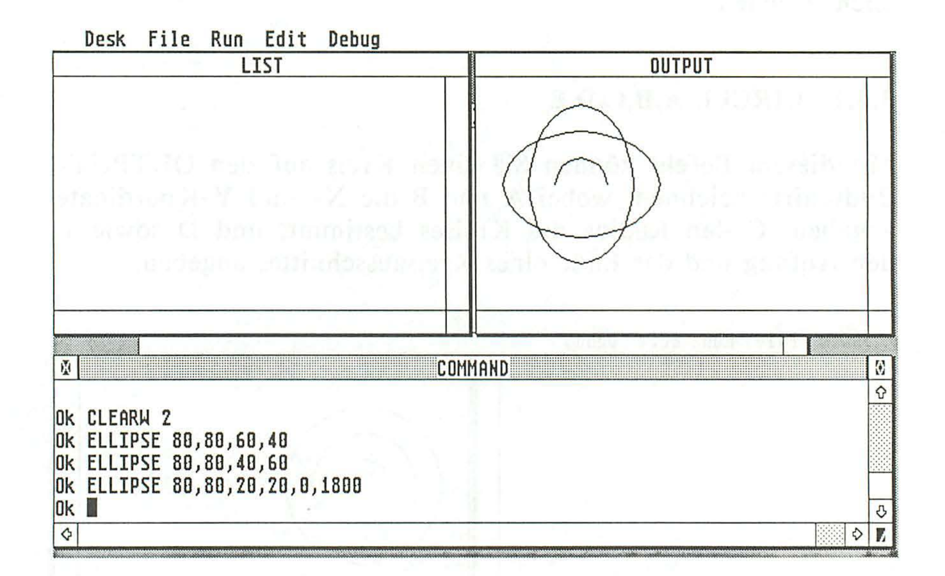
3.6.1 CIRCLE A,B,C,D,E

Mit diesem Befehl können Sie einen Kreis auf den OUTPUT-Bildschirm zeichnen, wobei A und B die X- und Y-Koordinate angeben, C den Radius des Kreises bestimmt, und D sowie E den Anfang und das Ende eines Kreisausschnittes angeben.



3.6.2 ELLIPSE A,B,C,D,E,F

Dieser Befehl bewirkt das Zeichnen einer Ellipse auf dem OUTPUT-Bildschirm, wobei A und B die X- und Y-Koordinate des X-Radius angeben, C und D den Y-Radius bestimmen und E sowie F den Anfang und das Ende eines Ellipsenausschnittes angeben.



3.6.3 OPENW A

Befehl zum Öffnen der einzelnen Windows (0=EDIT, 1=LIST, 2=OUTPUT, 3=COMMAND).

3.6.4 CLOSEW A

Befehl zum Schließen der einzelnen Windows (0=EDIT, 1=LIST, 2=OUTPUT, 3=COMMAND).

3.6.5 FULLW A

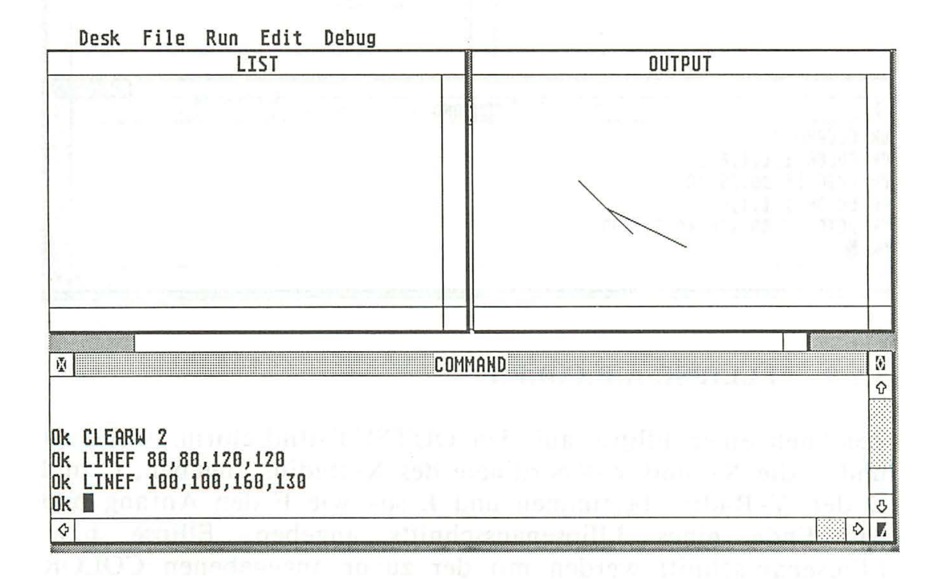
Befehl zum Öffnen der einzelnen Windows auf die volle Bildschirmgröße (0=EDIT, 1=LIST, 2=OUTPUT, 3=COMMAND).

3.6.6 GOTOXY A,B

Der Textcursor wird an die Bildschirmkoordinate A,B positioniert.

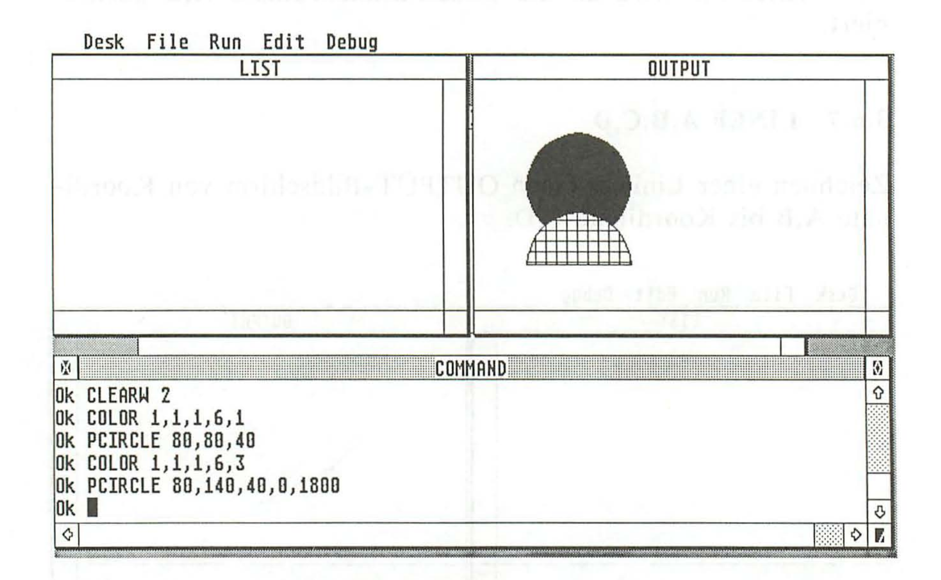
3.6.7 LINEF A,B,C,D

Zeichnen einer Linie auf den OUTPUT-Bildschirm von Koordinate A,B bis Koordinate C,D.



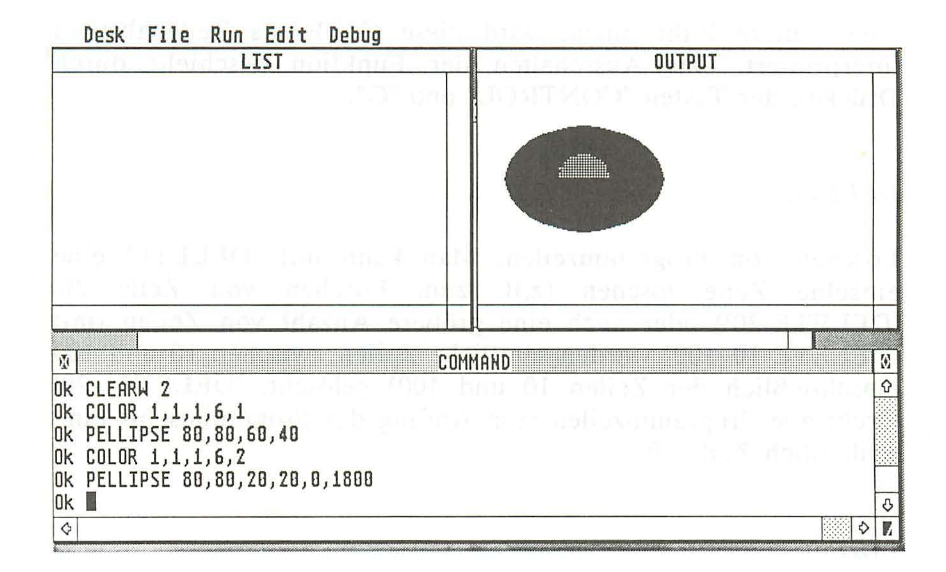
3.6.8 PCIRCLE A,B,C,D,E

Zeichnen eines Kreises auf den OUTPUT-Bildschirm, wobei A und B die X- und Y-Koordinate angeben, C den Radius des Kreises bestimmt und D sowie E den Anfang und das Ende eines Kreisausschnittes angeben. Kreis bzw. Kreisausschnitt werden mit der zuvor angegebenen COLOR ausgemalt. (COLOR Textfarbe,Füllfarbe,Linienfarbe,Style,Index)



3.6.9 PELLIPSE A,B,C,D,E,F

Zeichnen einer Ellipse auf den OUTPUT-Bildschirm, wobei A und B die X- und Y-Koordinate des X-Radius angeben, C und D den Y-Radius bestimmen und E so- wie F den Anfang und das Ende eines Ellipsenausschnitts angeben. Ellipse bzw. Ellipsenausschnitt werden mit der zuvor angegebenen COLOR ausgemalt.



3.7 Nützliche Helfer

3.7.1 Programmierhilfen

Die folgenden Befehle leisten gute Dienste beim Programmieren. Sie müssen ohne Zeilennummern im Direktmodus eingegeben werden.

AUTO

Nach Eingabe dieses Befehls wird beim Programmieren eine automatische Zeilennumerierung vorgegeben, d.h. Sie brauchen nicht mehr selbst die Zeilennummern in aufsteigender Reihenfolge einzugeben; nach Drücken der Eingabetaste wird automatisch die nächste Zeilennummer angezeigt. Setzen Sie eine Zahl hinter den 'AUTO'-Befehl, wird mit dieser Zahl = Zeilennummer die automatische Zeilennummernausgabe eingeleitet ('AUTO 100' bedeutet demnach: die automatische Numerierung beginnt mit Zeile 100). Setzen Sie zudem noch ein Komma und

eine weitere Zahl hinzu, wird diese Größe als Zeilenabstand interpretiert. Das Ausschalten der Funktion geschieht durch Drücken der Tasten "CONTROL" und "G".

DELETE

Löschen von Programmzeilen. Man kann mit 'DELETE' eine einzelne Zeile löschen (z.B. zum Löschen von Zeile 20: 'DELETE 20') oder auch eine größere Anzahl von Zeilen (mit 'DELETE 10-100' werden sämtliche Zeilen zwischen 10 und 100 einschließlich der Zeilen 10 und 100) gelöscht. 'DELETE -20' löscht alle Programmzeilen vom Anfang des Programms bis einschließlich Zeile 20.

LIST

Auflistung des Programms auf dem Bildschirm. Man kann mit 'LIST' auch eine ganz bestimmte Zeile auf dem Bildschirm zeigen lassen, indem man die Zeilennummer hinzufügt z.B.: 'LIST 10' zeigt nur Zeile 10 an; 'LIST 10-100' zeigt die Zeilen zwischen 10 und 100 auf dem Bildschirm an (einschließlich der Zeilen 10 und 100).

RENUM

Nachträgliche Umnumerierung der Zeilen im Speicher (RENUM = Abk. von renumber = umnumerieren). Man kann diesen Befehl um drei Werte ("Parameter") erweitern: 'RENUM 10,100,5' bedeutet: Numeriere das Programm ab Zeile 100 neu, starte aber mit Zeilennummer 10 und wähle einen Zeilenabstand von jeweils 5.

3.7.2 Wanzenkiller

Hier eine kurze Übersicht über die Hilfen zum "Debuggen" (Befreien von Fehlern; bug (engl.) = Wanze)

1. Unterbrechen des laufenden Programms durch 'Control'-'G' und Fortsetzung durch 'CONT'.

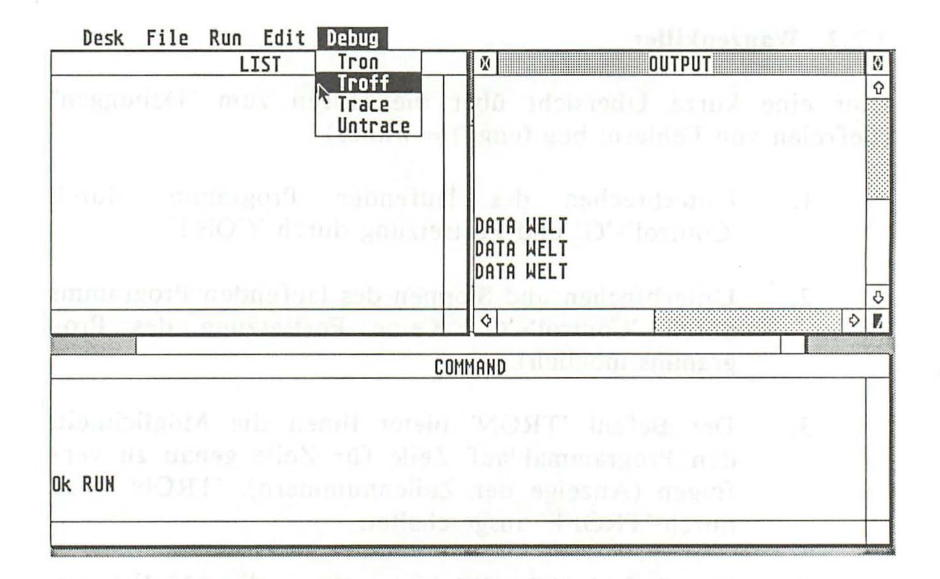
2. Unterbrechen und Stoppen des laufenden Programms durch 'Control'-'C'. Keine Fortsetzung des Programms möglich).

3. Der Befehl 'TRON' bietet Ihnen die Möglichkeit, den Programmablauf Zeile für Zeile genau zu verfolgen (Anzeige der Zeilennummern). 'TRON' wird durch 'TROFF' ausgeschaltet.

4. Der Befehl 'TRACE' bietet Ihnen die Möglichkeit, den Programmablauf Zeile für Zeile genau zu verfolgen (im Unterschied zu TRON Anzeige der gesamten Zeile). 'TRACE' wird durch 'UNTRACE' ausgeschaltet.

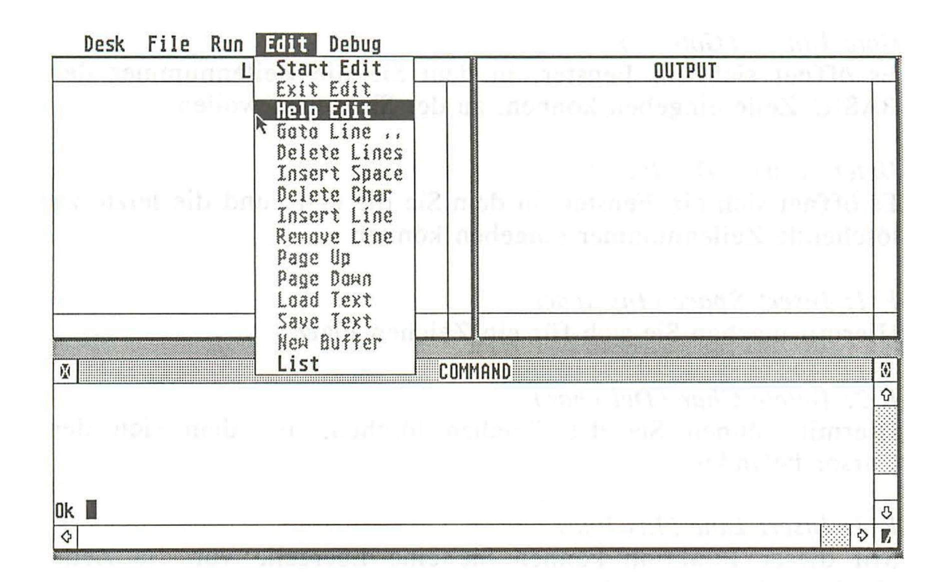
5. Der Befehl 'FOLLOW A' zeigt Ihnen während des Programmablaufs den z.Z. aktuellen Wert der Variablen A auf dem 'COMMAND'-Bildschirm an.

Die Befehle TRACE, UNTRACE, TRON und TROFF finden Sie im "Debug"-Menü, Sie können die Befehle aber natürlich auch per Tastatur eingeben.



3.7.3 EDIT hilft beim "Debuggen"

Sicher haben Sie sich schon gefragt, warum wir bisher das EDIT-Fenster des BASIC so sträflich vernachlässigt haben. Das EDIT-Fenster ist eine Art Schmierpapier, auf dem Sie nach Herzenslust an Ihrem Programm herumkorrigieren können. Sie rufen das EDIT-Fenster und damit den EDIT-Modus entweder durch die Tastatureingabe "EDIT" oder Anklicken des "Start Edit"-Feldes im EDIT-Menü auf. Auch z.B. "EDIT 20" ist möglich, nur Zeile 20 wird im EDIT-Fenster angezeigt. Im EDIT-Menü finden Sie auch eine ganze Liste von Befehlen, mit denen es sich hervorragend arbeiten läßt:



In der neuen BASIC-Version wird hinter den einzelnen Menüpunkten noch die zum gleichen Ergebnis führende Funktionstastenbelegung angezeigt. Beim alten wie beim neuen BASIC können Sie sich diese aber auch mit 'Help Edit' anzeigen lassen.

Hier nun die Beschreibung der Befehle von oben nach unten. Die Angaben in Klammern beziehen sich auf das neue BASIC.

Start Edit (Edit)
Hiermit wird der BASIC-Editor gestartet.

F10: Exit Edit (Exit)
Hiermit wird der BASIC-Editor wieder verlassen.

Help Edit (Help)
Sie erhalten eine Übersicht über die Funktionstastenbelegung des Editors (s. nächstes Bild).

Goto Line .. (Goto ...)
Es öffnet sich ein Fenster, in dem Sie die Zeilennummer der BASIC-Zeile eingeben können, zu der Sie gehen wollen.

Delete Lines (Del lines)
Es öffnet sich ein Fenster, in dem Sie die erste und die letzte zu löschende Zeilennummer eingeben können.

F 1: Insert Space (Ins space)
Hiermit machen Sie sich für ein Zeichen Platz.

F 2: Delete Char (Del char)
Hiermit können Sie das Zeichen löschen, auf dem sich der Cursor befindet.

F 3: Insert Line (Ins line)
Mit dieser Funktion können Sie eine Leerzeile vor die Zeile einfügen, in der der Cursor steht.

F 4: Remove Line (Del line)
Diese Funktion löscht die Zeile, in der der Cursor steht. Die Zeile erscheint danach grau und ist nicht mehr im Programm enthalten. Sie kann wieder übernommen werden, wenn man mit dem Cursor auf die Zeile fährt und die Eingabetaste betätigt.

F 5: Page Up (Page Up)
Mit dieser Funktion blättern Sie eine Bildschirmseite zurück, um den Programmtext zu sehen, der vor den sichtbaren Zeilen liegt.

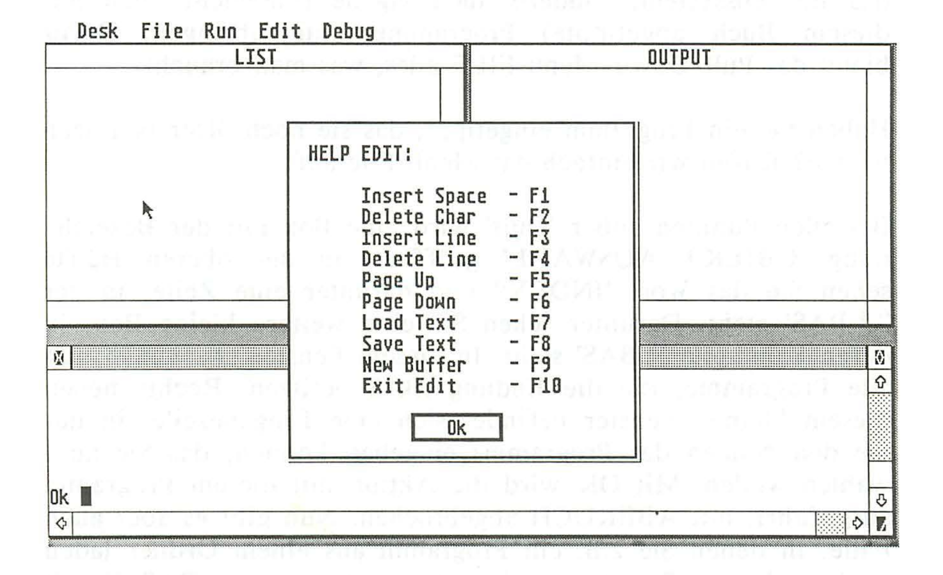
F 6: Page Down (Page Down)
Hiermit blättern Sie eine Seite vor.

F 7: Load Text (Load Text)
Bei dieser Funktion wird ein Text geladen, der mit der Funktion 'Save Text' gespeichert wurde. Bitte benutzen Sie möglichst nicht 'Load Text' und das folgende 'Save Text', da diese Funktionen lediglich Teile des Programms abspeichern bzw. laden und Sie versehentlich wichtige Teile nicht abgespeichert haben. Benutzen Sie sicherheitshalber die normalen 'LOAD' und 'SAVE AS' Befehle unter dem Menüpunkt 'FILE'.

F 8: Save Text (Save Text)
Hiermit wird der Text auf Diskette zwischengespeichert und kann mit 'Load Text' wieder geladen werden.

F 9: New Buffer (New Buf)
Mit dieser Funktion löschen Sie endgültig die grau unterlegten BASIC-Zeilen aus Ihrem Programm

List (List)
Ihr Programm wird im List-Fenster aufgelistet.



Das Editieren ist nun sehr einfach. Nach Aufruf des EDIT-Fensters findet man das Programm dort ebenfalls aufgelistet. Jetzt kann man zeilenweise herumkorrigieren. Das Geänderte wird dabei stets grau angezeigt. Bevor man nun die geänderte Zeile mit der Eingabetaste bestätigt, kann man sich sein Werk ansehen, abspeichern (Save Text) und auch bearbeiten, wobei dann die oben genannten Funktionen benutzt werden können. Am besten schreiben Sie einmal ein kleines Programmchen

(vielleicht einfach eins aus den vorigen Abschnitten abtippen) und experimentieren mit den Editiermöglichkeiten etwas herum. Sie werden sehen, auch hier ist der ST wieder sehr komfortabel.

3.8 Arbeiten mit dem Diskettenlaufwerk

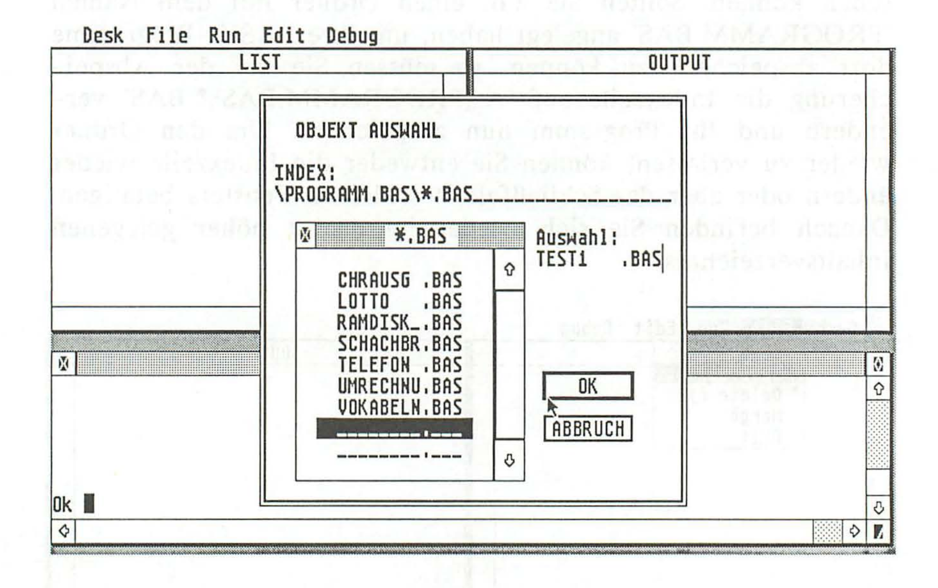
Erinnern Sie sich noch? Vorhin haben wir zusammen gelernt, daß man eine Diskette vor dem Gebrauch immer erst formatieren muß. Auf einer solchen formatierten Diskette hat man nun die Möglichkeit, nicht nur die Programmiersprachen und das Betriebssystem, sondern auch eigene (vielleicht auch aus diesem Buch abgetippte) Programme unterzubringen. Dafür bietet das Pull-Down-Menü FILE alles, was man braucht.

Haben Sie ein Programm eingetippt, das sie noch öfter benutzen wollen? Rufen wir einfach das Menü File auf.

Bei allen Punkten außer 'Quit' wird eine Box mit der Bezeichnung 'OBJEKT AUSWAHL' geöffnet. In der oberen Hälfte sehen Sie das Wort 'INDEX:' und darunter eine Zeile, in der '*.BAS' steht. Darunter sehen Sie eine weitere kleine Box, in deren Kopfzeile '*.BAS' steht. In diesem Fenster erscheinen nun alle Programme, die die Endung .BAS besitzen. Rechts neben diesem kleinen Fenster befindet sich eine Eingabezeile, in der Sie den Namen des Programms eingeben können, das Sie auswählen wollen. Mit OK wird die Aktion mit diesem Programm ausgeführt, mit ABBRUCH abgebrochen. Nun gibt es aber auch Fälle, in denen Sie z.B. ein Programm aus einem Ordner laden wollen oder ein Programm mit anderem Anhang als .BAS. Damit Sie auch diese Programme sehen können, müssen Sie den Index verändern. Bewegen Sie dazu den Mauspfeil auf die Zeile unterhalb von Index. Nun sehen Sie, wie der senkrechte Strich, der für den Eingabecursor steht, hinter dem letzten Zeichen in dieser Zeile erscheint. Löschen Sie die Zeile durch die Taste 'Esc' und tragen Sie z. B. 'A:*.*' ein, um alle Programme der Diskette zu sehen. Damit Ihre Anweisung auch ausgeführt wird, müssen Sie mit der Maus die Titelzeile des kleinen Fensters anklicken. Betätigen Sie bitte nicht die Eingabetaste, da das gesamte Fenster sonst sofort wieder geschlossen wird. Jetzt sollten

Sie sämtliche Programme, die sich auf der Diskette befinden, sehen können. Sollten Sie z.B. einen Ordner mit dem Namen 'PROGRAMM.BAS' angelegt haben, um Ihre BASIC-Programme dort abspeichern zu können, so müssen Sie bei der Abspeicherung die Indexzeile auf 'A:\PROGRAMM.BAS*.BAS' verändern und Ihr Programm nun abspeichern. Um den Ordner wieder zu verlassen, können Sie entweder die Indexzeile wieder ändern oder aber das Schließfeld des kleinen Fensters betätigen. Danach befinden Sie sich wieder im nächst höher gelegenen Inhaltsverzeichnis.

3.8.1 LOAD und SAVE



Haben Sie eine formatierte, neue Diskette zur Hand? Dann klicken wir nun "SAVE AS" (Speichern als) an. Nun können wir einen Namen für unser Programm eintippen. Tippen Sie z.B. TEST1, und unter dem Wort "Auswahl" erscheint der Name. Jetzt einfach das OK-Feld angeklickt, und Ihr Programm wird gespeichert und ist nun jederzeit unter diesem Namen für Sie abrufbar.

Wie? Die Funktion LOAD (= Laden) läßt wieder das obige, vertraute Bild erscheinen. Ihr Programm ist nun links in dem Fenster zu sehen. Haben Sie bereits mehrere Programme abgespeichert, wählen Sie mit der Maus das entsprechende Programm aus, klicken an, und es erscheint im "Auswahl" Feld. Wenn Sie nun das OK-Feld anklicken, wird Ihr Programm in den Speicher geladen.

3.8.2 DELETE FILE, MERGE und QUIT

Mit *DELETE FILE* (File löschen) können wir ein zuvor gespeichertes Programm von der Diskette löschen. Diese Funktion entspricht dem Schieben eines Bildchens über den Papierkorb. Wiederum erscheint das bekannte Bild, und mit der Maus wird das zu löschende File angeklickt, erscheint unter "Auswahl" und wird nach OK-Bestätigung gelöscht.

MERGE ermöglicht es, ein noch auf der Diskette befindliches Programm mit dem zur Zeit im Speicher stehenden Programm zu vermischen. Vorsicht mit diesem Befehl!!! Nach dem Ausführen der Funktion haben Sie nur noch ein Programm, das beide Einzelprogramme in sich enthält.

Es ist so möglich, ein Programm an das andere anzuhängen. Beispiel: Programm 1 hat die Zeilennummern von 10 bis 150. Das zweite Programm muß nun bei Zeile 160 anfangen, sonst hat man hinterher einen Programmbrei, der sich schwer wieder entknoten läßt.

QUIT bewirkt das Verlassen von BASIC und die Rückkehr zum GEM Desktop.

So, nun haben wir ein paar der wichtigsten Befehle des fantastischen ST-BASIC kennengelernt. Versuchen Sie sich selbst einmal ein wenig damit. Natürlich konnten wir bei weitem nicht alle Befehle behandeln. Haben Sie nun Geschmack an BASIC gefunden, sollten Sie sich in anderen Büchern weiter informieren. Im Rahmen dieses Einsteigerbuches soll es zunächst genug sein.

3.9 Erste Programme

Bis hierher alles klar? Dann geht es jetzt richtig los. Auf den folgenden Seiten finden Sie BASIC-Programme, die sich fast voll und ganz aus dem Vokabelschatz zusammensetzen, den wir in den vorangegangenen Abschnitten kennengelernt haben. Auf das eigentliche Programmlisting folgt dann jeweils eine ausführliche Erklärung der Zeilen.

An manchen Stellen werden Sie feststellen, daß Ihnen dieser oder jener Befehl vorher in diesem Buch noch nicht begegnet ist. In diesem Fall sind wir im Erläuterungstext zum Programm dann entsprechend tiefer darauf eingegangen.

Für jemanden, der BASIC als seine alltägliche Programmiersprache bezeichnet, bieten die abgedruckten Listings nichts Besonderes.

Aber das Problem bei langen Programmen besteht vor allem darin, daß die Computerneulinge sich ohne jegliches Wissen von BASIC (das Sie ja nach dem Studieren dieses Buches bereits in größeren Zügen vorweisen können) an seitenlange Listings herantrauen, ohne auch nur einen Bruchteil davon zu verstehen, was sie dort mühsam über mehrere Seiten abschreiben müssen. Kein Wunder, daß dann die Fehlersuche fast unmöglich wird, denn - davor sind auch Sie nicht gefeit - Tippfehler begehen wir alle. Nur der Unterschied beim Abtippen von Programmlistings ist halt der, daß ein Computer auch nicht den kleinsten Tippfehler verzeiht.

Noch zwei Hinweise:

1. Wenn Sie die folgenden Programme in Ihren ATARI ST eingeben, vergessen Sie bitte nicht, am Ende jeder Zeile die Eingabetaste zu betätigen.

2. Bei den Erläuterungen der Programme Zeile für Zeile haben wir die REM-Zeilen jeweils übersprungen.

3.9.1 Telefonbuch



Programm Nummer 1 soll Ihnen zeigen, wie Ihr ST als Telefonbuch eingesetzt werden kann.

Das Programm läuft folgendermaßen ab:

Erst werden Sie nach der Anzahl der einzugebenden Teilnehmer gefragt. Dann geben Sie die Namen und Telefonnummern ein. Jetzt können Sie dem ST einen Namen eingeben, und er findet Ihnen die zugehörige Telefonnummer.

```
10 REM Telefonbuch
20 CLEARW 2
30 INPUT "Wie viele Telefonnummern ";A
```

```
40 REM Dimensionierung für Nummern und Namen
50 DIM NA$(A),NU$(A)
60 CLEARW 2
70 REM Aufbau des Telefonverzeichnisses
80 FOR N=1 TO A
90 INPUT "Name ";NA$(N)
100 INPUT "Nummer ";NU$(N)
110 PRINT
120 NEXT N
130 REM Suche von Telefonnummern
140 CLEARW 2
150 INPUT "Name bitte eingeben";A$
160 FOR N=1 TO A
170 REM Name gefunden, so Ausgabe der Nummer
180 IF A$=NA$(N) THEN GOSUB 210
190 NEXT N
200 GOTO 140
210 REM Unterprogramm: Ausgabe der Nummer
220 PRINT "Die Telefonnummer: ";NU$(N)
230 PRINT
240 INPUT "Bitte Eingabetaste drücken ";B$
250 RETURN
```

Nun schauen wir uns das Programm Zeile für Zeile an:

Zeile 20: Der OUTPUT-Bildschirm wird gelöscht.

Zeile 30: In der Schublade A wird die Anzahl der im folgenden einzugebenden Telefonnummern und somit auch die Anzahl der dazugehörenden Gesprächsteilnehmer abgelegt.

Zeile 50: Hier wird nun nicht nur die Anzahl der Variablen für die Telefonnummer (die Zahlvariable NU(A)), sondern auch die der Teilnehmer (die Stringvariable NA$(A)) festgelegt. Obgleich die Telefonnummer eine Zahl ist, mußten wir wegen des Querstriches (0251/47478) und der ersten Stelle (0) eine Stringvariable wählen.

Zeile 60: Erneutes Löschen des OUTPUT-Bildschirms.

Zeile 70-120: Abfrage des Teilnehmernamens (Zeile 90) und seiner Telefonnummer (Zeile 100).

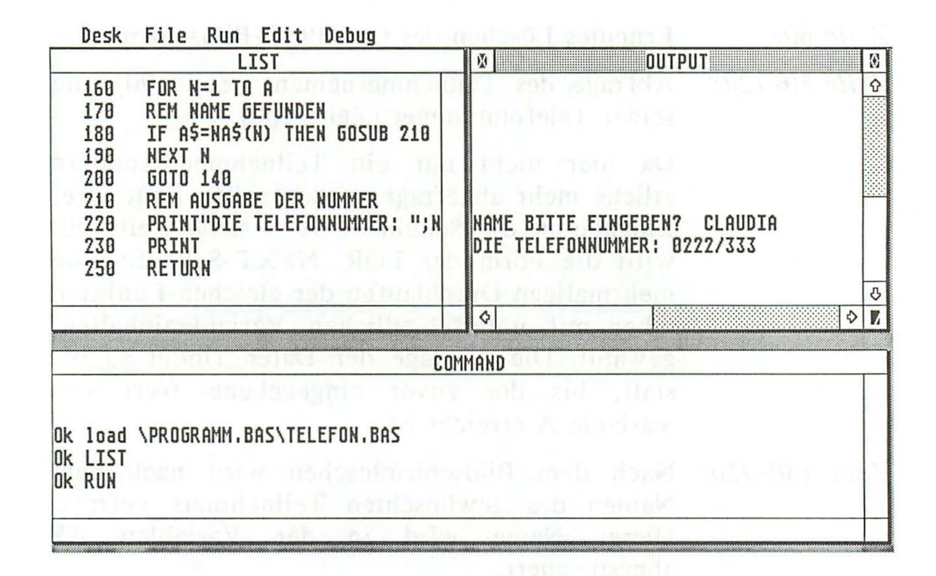
Da hier nicht nur ein Teilnehmer, sondern etliche mehr abgefragt werden sollen (laut Ihrer Eingabe in die Schublade A - siehe Zeile 30) wird die Form der FOR...NEXT-Schleife zum mehrmaligen Durchlaufen der gleichen Funktion (aber mit unterschiedlichen Variableninhalten) gewählt. Die Abfrage der Daten findet so oft statt, bis der zuvor eingegebene Wert von Variable A erreicht ist.

Zeile 130-150: Nach dem Bildschirmlöschen wird nach dem Namen des gewünschten Teilnehmers gefragt. Dieser Name wird in der Variablen A$ abgespeichert.

Zeile 160-190: Nun werden erneut mit einer FOR...NEXT-Schleife sämtliche bereits abgespeicherten Namen untersucht. Wenn einer dieser Namen (NA$(N)) dem Namen des gewünschten Teilnehmers zu 100% entspricht, verzweigt das Programm ins Unterprogramm in Zeile 210ff.

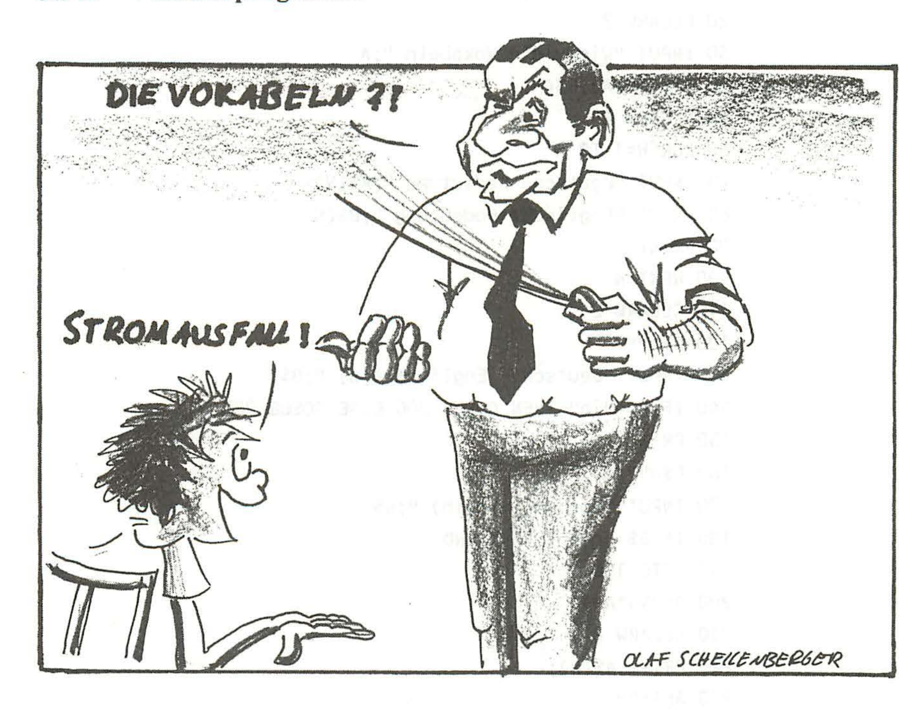
Wenn hingegen die Suche erfolglos verlief, läßt der Programmschritt in Zeile 200 die gleiche Abfrageprozedur wieder neu starten.

Zeile 210-250: In dieses Unterprogramm wird nur dann verzweigt, wenn bereits die Übereinstimmung der Variablen A$ (gewünschter Teilnehmer) und NA$(N) (abgespeicherter Teilnehmer) eindeutig festgestellt wurde. In diesem Fall wird in Zeile 220 die Ausgabe der gewünschten Telefonnummer veranlaßt.



Zeile 240 erfüllt die spezielle Funktion, daß erst nach Drücken der Eingabetaste mit dem weiteren Programmablauf fortgefahren wird. Sonst könnte es passieren, daß Ihr ST zu schnell wieder ins Hauptprogramm zurückkehrt und dabei den OUTPUT- Bildschirm löscht (Zeile 140), ohne daß Sie zuvor die Ausgabe der gewünschten Telefonnummer auf dem Bildschirm mitverfolgen konnten.

3.9.2 Vokabelprogramm



Dieses Programm läßt Ihren ST zum Vokabeltrainer werden.

Zuerst werden Sie nach der Anzahl der zu speichernden Vokabeln gefragt und anschließend dazu aufgefordert, die englische und die dazu passende deutsche Bedeutung einzutragen.

Danach können Sie auswählen, ob Ihnen der ST die deutsche oder die englische Vokabel zur Abfrage vorlegen soll.

Schließlich sucht er Ihnen zufällig Vokabeln heraus, die Sie daraufhin möglichst richtig in der anderen Sprache eingeben sollten.

```
10 REM Vokabelprogramm
20 CLEARW 2
30 INPUT "Wie viele Vokabeln ";A
40 DIM A$(A),B$(A)
50 CLEARW 2
60 FOR N=1 TO A
70 INPUT "Deutsche Bedeutung ";A$(N)
80 INPUT "Englische Bedeutung ";B$(N)
90 PRINT
100 NEXT N
110 CLEARW 2
120 B$=""
130 INPUT "Deutsch - Englisch (ja) ";B$
140 IF B$="ja" THEN GOSUB 200 ELSE GOSUB 280
150 PRINT
160 B$=""
170 INPUT "Noch mal (nein) ";B$
180 IF B$="nein" THEN END
190 GOTO 110
200 B=RND*A+1
210 CLEARW 2
220 PRINT A$(B);
230 AN$=""
240 INPUT AN$
250 IF AN$=B$(B) THEN PRINT "Richtig!"
260 IF AN$<>B$(B) THEN PRINT "Falsch!"
270 RETURN
280 B=RND*A+1
290 CLEARW 2
300 PRINT B$(B);
310 AN$=""
320 INPUT AN$
330 IF AN$=A$(B) THEN PRINT "Richtig!"
340 IF AN$<>A$(B) THEN PRINT "Falsch!"
350 RETURN
```

Nun schauen wir uns das Programm Zeile für Zeile an:

Zeile 20: Löschen des OUTPUT-Bildschirms.

Zeile 30: In der Schublade A wird die Anzahl der im folgenden einzugebenden Vokabeln und somit auch die Anzahl der dazugehörenden Übersetzungen abgelegt.

Zeile 40: Hier wird nun nicht nur die Anzahl der Stringvariablen für die deutschen Bedeutungen (A$(A)), sondern auch die der englischen Bedeutungen (B$(A)) festgelegt.

Zeile 50: Erneutes Löschen des OUTPUT-Bildschirms.

Zeile 60-100: Abfrage der deutschen Bedeutung (Zeile 70) und der englischen Bedeutung (Zeile 80). (Selbstverständlich können Sie überall statt "Englisch" jede beliebige Fremdsprache einsetzen.)

Da wir hier nicht nur eine Vokabel, sondern etliche mehr abgefragt werden sollen (laut Ihrer Eingabe in die Schublade A - siehe Zeile 30), wird die FOR...NEXT-Schleife zum mehrmaligen Durchlaufen der gleichen Funktion (aber mit unterschiedlichen Variableninhalten) gewählt. Die Abfrage der Daten findet so oft statt, bis der Wert von Variabe A erreicht ist.

Zeile 110-130: Nach dem Löschen des Bildschirms werden Sie gefragt, ob Sie Deutsch - Englisch ('ja' eingeben) oder ob Sie lieber Englisch - Deutsch abgefragt werden wollen (in diesem Fall etwas anderes als 'ja' eingeben). Ihre Antwort wird in der Stringvariablen B$ abgespeichert.

Zeile 140: Hier wird nun entschieden, in welches Unterprogramm Ihr ST verzweigen soll. Haben Sie in Zeile 130 mit 'ja' geantwortet, dann springt Ihr ST in das Unterprogramm in Zeile 190 (Abfrage Deutsch - Englisch), andernfalls (Abfrage Englisch - Deutsch wird gewünscht) springt er in das Unterprogramm in Zeile 260.

Wie Sie sehen, ist hier unser bereits bekannter Befehl IF...THEN...ELSE. Wir können uns die Programmzeile 140 folgendermaßen ins Deutsche übersetzen: Wenn B$="ja", dann springe ins Unterprogramm in Zeile 190, andernfalls springe ins Unterprogramm in Zeile 260.

Bei unserer wenn...dann-Abfrage gibt es immer zwei Möglichkeiten: Entweder ist die wenn-Aussage richtig, dann wird direkt der Befehl hinter dem Wörtchen wenn ausgeführt, ist dies aber nicht der Fall, wird der Befehl hinter dem Wörtchen ELSE (andererseits) ausgeführt. Fehlt ELSE in einer IF-Anweisung und trifft die Aussage nicht zu, wird das Programm in der nächsten Programmzeile fortgesetzt.

Zeile 150-190: Nach der Vokabelabfrage wird der Übersichtlichkeit halber eine Leerzeile auf dem Bildschirm erzeugt (Zeile 150), nachgefragt, ob weiteres Vokabellernen erwünscht wird (Zeile 170) und aufgrund dieser Eingabe entweder das Programm beendet (Zeile 180 - der neue Befehl END besagt: Hier hat der Computer die Ausführung zu unterbrechen) oder ob mit der Abfrage fortgefahren werden soll (Zeile 190).

```
Desk  File  Run  Edit  Debug

LIST
 270   RETURN
 280   B=RND*A+1
 290   CLEARW 2
 300   PRINT B$(B);
 310   AN$=""
 320   INPUT AN$
 330   IF AN$=A$(B) THEN PRINT "RIC
 340   IF AN$<>A$(B) THEN PRINT "FA
 350   RETURN

OUTPUT
DEUTSCHE BEDEUTUNG?  HUND
ENGLISCHE BEDEUTUNG?  DOG

DEUTSCHE BEDEUTUNG?  KATZE
ENGLISCHE BEDEUTUNG?  CAT

DEUTSCHE BEDEUTUNG?  JUNGE

COMMAND
Br stop
Ok NEW
Ok load \PROGRAMM.BAS\VOKABELN.BAS
Ok LIST
Ok RUN
```

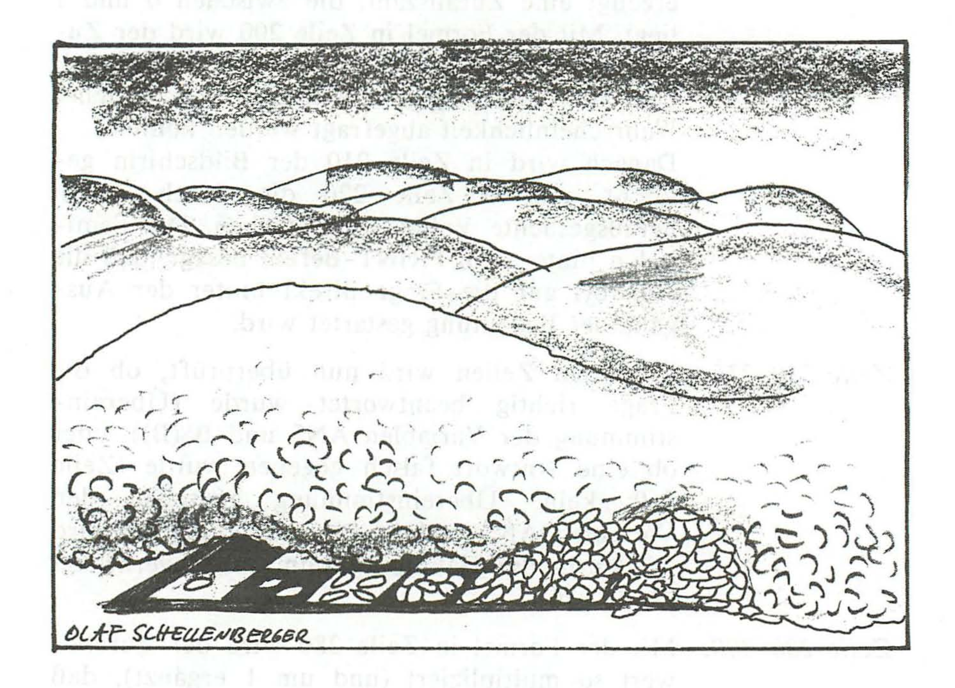
Zeile 200-220: Wieder taucht ein neuer Befehl auf: RND. Er erzeugt eine Zufallszahl, die zwischen 0 und 1 liegt. Mit der Formel in Zeile 200 wird der Zufallswert so multipliziert (und um 1 ergänzt), daß alle eingetragenen Vokabeln mit gleicher Wahrscheinlichkeit abgefragt werden können. Danach wird in Zeile 210 der Bildschirm gelöscht und in Zeile 220 die durch Zufall herausgesuchte Vokabel ausgegeben. Das Semikolon hinter dem PRINT-Befehl besagt, daß die Antwort auf die Frage direkt hinter der Ausgabe der Bedeutung gestartet wird.

Zeile 230-270: In diesen Zeilen wird nun überprüft, ob die Frage richtig beantwortet wurde (Übereinstimmung der Variablen AN$ und B$(B)), oder ob eine Antwort falsch gegeben wurde (Zeile 260: keine Übereinstimmung zwischen den Variablen AN$ und B$(B)). Anschließend wird wieder in das Hauptprogramm zurückverzweigt (Zeile 270).

Zeile 280-300: Mit der Formel in Zeile 280 wird der Zufallswert so multipliziert (und um 1 ergänzt), daß alle eingetragenen Vokabeln mit gleicher Wahrscheinlichkeit abgefragt werden können. Danach wird in Zeile 290 der Bildschirm gelöscht und in Zeile 300 die durch Zufall herausgesuchte Vokabel ausgegeben. Das Semikolon hinter dem PRINT-Befehl besagt, daß die Antwort auf die Frage direkt hinter der Ausgabe der Bedeutung gestartet wird.

Zeile 310-350: Hier wird überprüft, ob die Frage richtig beantwortet wurde, (Übereinstimmung der Variablen AN$ und A$(B)) oder ob eine falsche Antwort gegeben wurde (Zeile 340: keine Übereinstimmung zwischen den Variablen AN$ und A$(B)). Anschließend wird wieder in das Hauptprogramm zurückverzweigt (Zeile 350).

3.9.3 Das Schachbrettproblem



Programm Nummer 3 soll Ihnen zeigen, wie Ihr ST eine größere Rechenaufgabe in wenigen Sekunden lösen kann.

Das bekannte Schachbrettproblem stützt sich auf eine Geschichte, in der ein König einem Weisen die freie Wahl einer Belohnung gewährte. Der Weise wünschte sich daraufhin, ein normales Schachbrett mit Weizenkörnern nach folgender mathematischen Regel zu füllen: Die Weizenkörner sollten auf den 64 Feldern des Schachbretts nach folgender Regel verteilt werden: Auf das erste Feld 1 Weizenkorn, auf das zweite 2 Weizenkörner, auf das dritte Feld 4 Weizenkörner, auf das vierte Feld 8 Weizenkörner usw.

Auf den ersten Blick scheint das Problem sehr einfach zu lösen zu sein, denn wer weiß schon, wieviel Weizenkörner in einem Weizenbeutel stecken - da kommt es doch auf eins mehr oder weniger nicht mehr an?! Wenn Sie sich allerdings am Ende des Programmablaufes auf dem Bildschirm anschauen, wie groß die Zahl der Weizenkörner geworden ist, können Sie sich vielleicht vorstellen, daß diese Zahl noch nicht einmal mit Weizenkörnern ausgeglichen werden könnte, wie es sie heutzutage auf der ganzen Welt gibt. Der Weise hatte also einen unerfüllbaren Wunsch geäußert.

```
10 REM Schachbrettproblem
20 CLEARW 2
30 REM 64 Felder hat das Schachbrett
40 FOR N=1 TO 64
50 PRINT "Feld ";N
60 REM Potenzformel
70 A=2^(N-1)
80 PRINT A;"Weizenkörner"
90 REM In Schublade B liegt aller Weizen
100 B=B+A
110 PRINT "Insgesamt ";B;" Weizenkörner"
120 PRINT
125 FOR I=1 to 500: NEXT I
130 NEXT N
```

Die Zahl der Weizenkörner wird am Ende so groß sein, daß bei "Feld 64" die Zahl vom ST bereits nicht mehr angezeigt werden kann. Die letzte Angabe finden Sie daher unter "Feld 63". Sie können sich dann leicht vorstellen, wie unvorstellbar groß die Zahl im Feld 64 sein würde. Und hier die Erläuterung:

Zeile 20: Löschen des OUTPUT-Bildschirms

Zeile 40: Hier wird eine FOR...NEXT-Schleife eröffnet, damit alle 64 Schachfelder nacheinander berechnet werden können.

Zeile 50: Diese Zeile dient zur Information des aufmerksamen Zuschauers, denn sie erzeugt die Ausgabe der gerade behandelten Feldnummer auf dem Bildschirm.

Zeile 70: Nach dieser Rechenformel wird die Anzahl der Weizenkörner pro Feld errechnet. Der Pfeil nach oben (^) ist das Potenzierungszeichen und ist auf Ihrer ST-Tastatur durch gleichzeitiges Niederdrücken der Tasten SHIFT und Raute (#) einzugeben.

Zeile 80: Hier wird die Anzahl der Weizenkörner ausgegeben, die sich auf dem zur Zeit aktuellen Feld befinden.

Zeile 100: In der Variablen B wird die Anzahl der Weizenkörner abgespeichert und der Wert der bereits mit Weizenkörnern belegten Felder dazuaddiert.

Zeile 110: Hier erfolgt die Ausgabe sämtlicher Weizenkörner auf dem Schachbrett.

```
Desk  File  Run  Edit  Debug

LIST
 50    PRINT"FELD ";N
 60    REM POTENZFORMEL
 70    A=2^(N-1)
 80    PRINT A;"WEIZENKRNER"
 90    REM
 100   B=B+a
 110   PRINT"INSGESAMT ";B;" WEIZEN
 120   print
 130   next n

OUTPUT
INSGESAMT  511  WEIZENKRNER

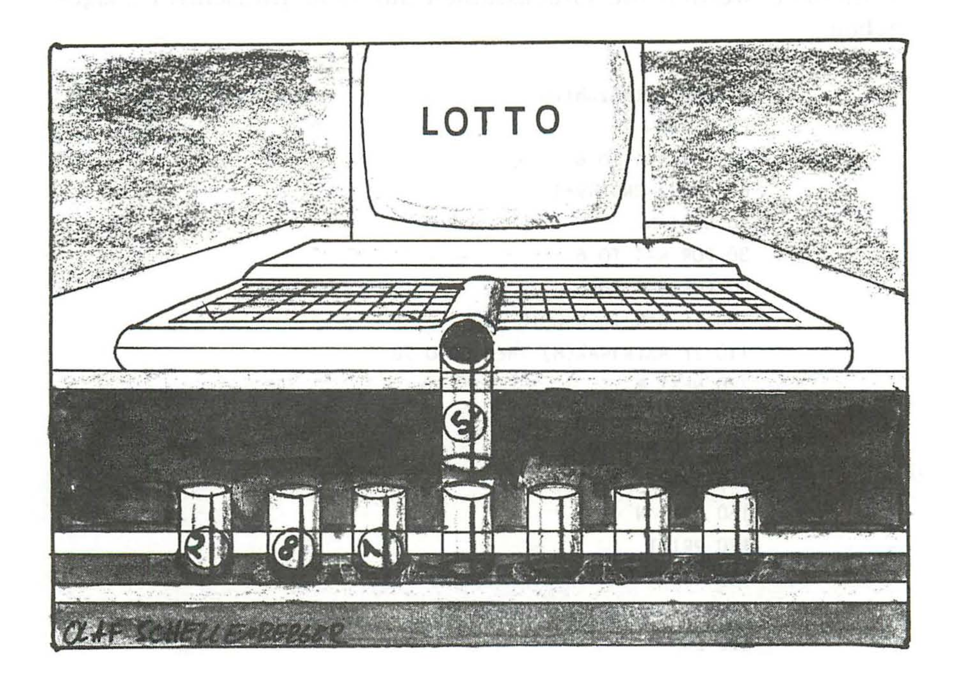
FELD  10
 512 WEIZENKRNER
INSGESAMT  1023  WEIZENKRNER
FELD  11
1024 WEIZENKRNER
INSGESAMT  2047  WEIZENKRNER

COMMAND
Ok load \PROGRAMM.BAS\SCHACHBR.BAS
Ok LIST

Ok LIST
Ok RUN
```

Zeile 120-130: In Zeile 120 wird der Übersichtlichkeit wegen eine Leerzeile auf dem OUTPUT-Bildschirm ausgegeben, während in Zeile 130 der Rücksprung zu Zeile 40 eingeleitet wird, um nach und nach alle 64 Schachbrettfelder zu berechnen. Zwischendurch wird in Zeile 125 durch die leere FOR...NEXT-Schleife eine Art Pause erzeugt, damit wir die Angaben auf dem Bildschirm ablesen können.

3.9.4 Lottozahlen



Programm Nummer 4 soll Ihnen zeigen, wie Ihr ST Ihnen dabei hilft (ohne Gewähr!), der Lottofee ein Schnippchen zu schlagen und die richtige Entscheidung für Samstag bzw. Mittwoch zur Auswahl der Lottozahlen zu unterstützen.

und die richtige Entscheidung für Samstag bzw. Mittwoch zur Auswahl der Lottozahlen zu unterstützen.

Auch in diesem Programm (wie bereits im Vokabelprogramm) wird auf die Zufallsfunktion (RND) der Programmiersprache BASIC zurückgegriffen.

Es werden 6 ausgewählt (gezogen), die allerdings vor der Bildschirmausgabe noch genau miteinander verglichen werden, damit nicht zwei oder mehr gleiche Zahlen ausgegeben werden. Anschließend werden die Glückszahlen auf dem Bildschirm ausgegeben.

```
10 REM Lottozahlen
20 CLEARW 2
50 FOR N=1 TO 6
60 A%(N)=RND*49+1
70 NEXT N
80 FOR N=1 TO 6
90 FOR M=1 TO 6
100 IF N=M THEN GOTO 120
110 IF A%(N)=A%(M) THEN GOTO 50
120 NEXT M
130 NEXT N
140 FOR N=1 TO 6
150 PRINT A%(N);
160 NEXT N
170 PRINT
180 INPUT " noch einmal (ja) ";A$
190 IF A$="ja" THEN GOTO 50
200 END
```

Nun schauen wir uns das Programm Zeile für Zeile etwas genauer an:

Zeile 10-20: Löschen des Bildschirms (Zeile 20).

Zeile 50-70: In diesen Zeilen werden nun die 6 Zufallszahlen für das Lotto erzeugt. Zur Abspeicherung haben wir Zahlvariablen gewählt, die nur ganze Zahlwerte annehmen können.

Die Multiplikation der Variablen A%(N) mit 49 geschieht selbstverständlich aus dem Grund, weil Zahlen zwischen 1 und 49 ausgewählt werden sollen.

Zeile 80-130: In diesem Zeilenblock werden sämtliche erzeugten Zufallszahlen miteinander vergleichen. Sind zwei Zufallszahlen gleich (Zeile 110), werden neue Zufallszahen erzeugt, indem in Zeile 50 zurückgesprungen wird.

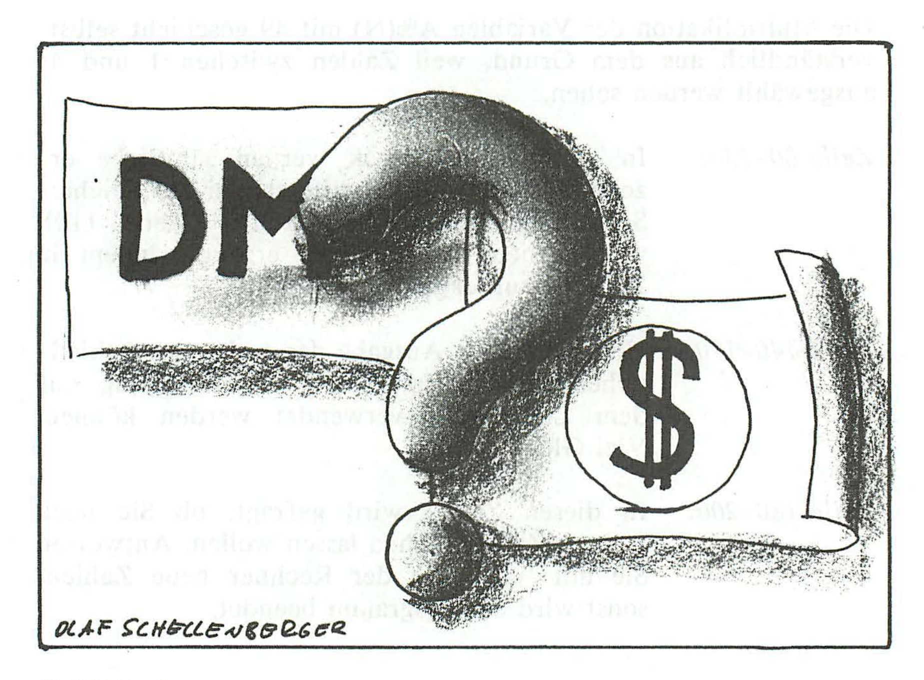
Zeile 140-170: Hier findet die Ausgabe der sechs unterschiedlichen Zahlen statt, die zur Eintragung auf dem Lottoschein verwendet werden können. Viel Glück!

Zeile 180-200: In diesen Zeilen wird gefragt, ob Sie noch weitere Zahlen ziehen lassen wollen. Antworten Sie mit 'ja', zieht der Rechner neue Zahlen, sonst wird das Programm beendet.

3.9.5 Umrechnungsprogramm

Programm Nummer 5 besteht eigentlich aus 4 einzelnen Unterprogrammen. In diesen Programmen wird Ihnen gezeigt, wie Ihr ST sich Umrechnungen merken kann: Von DM in Dollar, von Dollar in DM, von cm in inch und von inch in cm. Die Reihe ließe sich beliebig fortsetzen.

Sie können nun komfortabel in einem Menü auswählen, welches Unterprogramm gestartet werden soll. Nach der Eingabe der gewünschten Kennzahl und Drücken der Eingabetaste wird das entsprechende Unterprogramm aufgerufen.




```
10 REM Menü
15 CLEARW 2
20 PRINT "Bitte wählen Sie:"
30 PRINT "1 Umrechnung DM in Dollar"
40 PRINT "2 Umrechnung Dollar in DM"
50 PRINT "3 Umrechnung cm in inch"
60 PRINT "4 Umrechnung inch in cm"
70 INPUT "Welche Umrechnung wünschen Sie";X
75 REM Sprung zum entsprechenden Unterprogramm
80 ON X GOTO 90,150,210,270
90 REM Umrechnungsprogramm DM-Dollar
100 CLEARW 2
110 INPUT "Bitte DM eingeben ";A
120 B=A/2.67
130 PRINT "Das sind";B;"Dollar"
140 END
150 REM Umrechnungsprogramm Dollar-DM
160 CLEARW 2
170 INPUT "Bitte Dollar eingeben ";A
180 B=A*2.67
190 PRINT "Das sind";B;"DM"
200 END
210 REM Umrechnungsprogramm cm-inch
```

```
220 CLEARW 2
230 INPUT "Bitte cm eingeben ";A
240 B=A/2.54
250 PRINT "Das sind";B;"inch"
260 END
270 REM Umrechnungsprogramm inch-cm
280 CLEARW 2
290 INPUT "Bitte inch eingeben ";A
300 B=A*2.54
310 PRINT "Das sind";B;"cm"
320 END
```

Nun schauen wir uns das Programm Zeile für Zeile an:

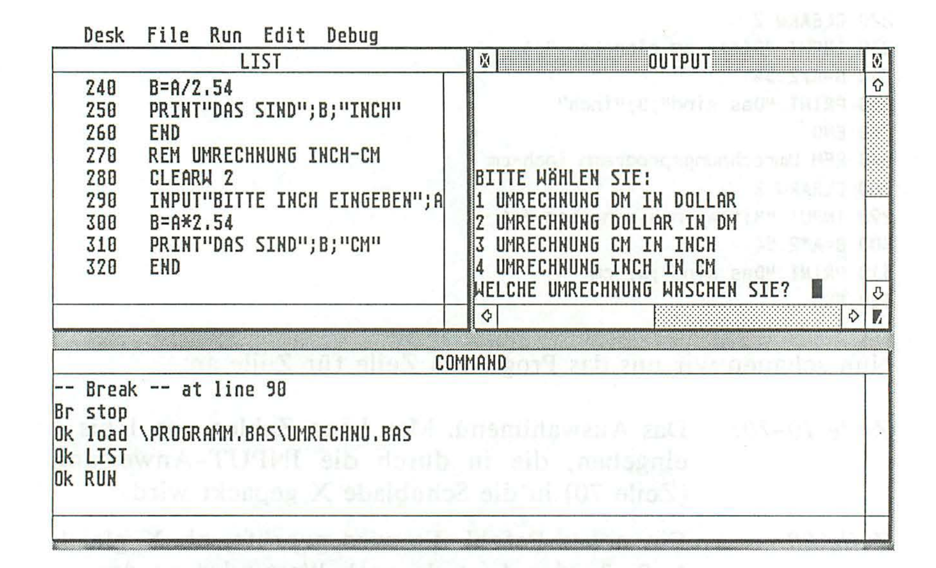
Zeile 10-70: Das Auswahlmenü. Man kann Zahlen von 1 bis 4 eingeben, die in durch die INPUT-Anweisung (Zeile 70) in die Schublade X gepackt wird.

Zeile 80: Ein neuer Befehl. Es wird geprüft, ob X gleich 1, 2, 3 oder 4 ist. Je nach Wert wird zu der 1., 2., 3. oder 4. hinter GOTO angegebenen Zeilennummer gesprungen.

Zeile 90-140: Nach dem Löschen des OUTPUT-Bildschirms (Zeile 100) wird nach dem gewünschten DM-Betrag gefragt (Zeile 110). Dieser Betrag wird durch den aktuellen Dollarstand - im Augenblick hat der Dollar den Wert 2.67 DM - geteilt (Zeile 120). Anschließend erfolgt in Zeile 130 die Ausgabe des Dollarwertes auf den OUTPUT-Bildschirm.

Zeile 150-200: Nach dem Löschen des OUTPUT-Bildschirms (Zeile 160) wird nach dem gewünschten Dollar-Betrag gefragt. Dieser Betrag wird mit dem aktuellen Dollarstand multipliziert (Zeile 180). Anschließend erfolgt in Zeile 190 die Ausgabe des DM-Betrags auf den OUTPUT-Bildschirm.

Zeile 210-260: Nach dem Löschen des OUTPUT-Bildschirms (Zeile 220) wird nach dem gewünschten cm-Wert gefragt (Zeile 230). Dieser Wert wird durch den aktuellen Inchmultiplikator (2.54) geteilt (Zeile 240). Anschließend erfolgt in Zeile 250 die Ausgabe des Inchwertes auf den OUTPUT-Bildschirm.



Zeile 270-320: Nach dem Löschen des Bildschirms (Zeile 280) wird nach dem gewünschten Inch-Wert gefragt (Zeile 290). Dieser Wert wird mit dem aktuellen Inchmultiplikator malgenommen (Zeile 300).

Anschließend erfolgt in Zeile 310 die Ausgabe des cm-Wertes auf den OUTPUT-Bildschirm.

3.10 Kurzübersicht ATARI ST BASIC-Grundwortschatz

CLEARW n: Löschen der einzelnen Bildschirmfenster:

n = 0: EDIT-Bildschirm
n = 1: LIST-Bildschirm
n = 2: OUTPUT-Bldschirm
n = 3: COMMAND-Bildschirm

CONT: Programmablauf wird nach Unterbrechung durch 'Control'-'G' oder das BASIC-Kommando 'STOP' fortgesetzt.

END: Anzeige des Programmendes. Nach diesem BASIC-Kommando kann das Programm nicht mit 'CONT' in seiner Ausführung fortgesetzt werden.

FOR...NEXT: Einleitung und Abschluß einer Programmschleife, die den Wert einer Variablen bei jedem Durchlauf um 1 erhöht.

GOTO: Sprung in eine bestimmte Programmzeile.

GOSUB...RETURN: Sprung in ein Unterprogramm und Rückkehr aus selbigem ins Hauptprogramm.

IF...THEN...ELSE: IF = wenn, THEN = dann, ELSE = anderenfalls.

INPUT: Einlesen von Daten in eine vorgegebene Variable; der Befehl INPUT kann auch zusätzlich zu Beginn eine Bildschirmausgabe enthalten.

LIST: Abbildung des Programms auf dem 'LIST'-Bildschirm.

NEW: Löschen des Programms im Speicher.

PRINT: Ausgabe von Variablen und Texten auf dem Bildschirm.

REM: Die Anweisungen hinter diesem Befehl gelten lediglich zur Information, sie werden nicht vom Computer weiter verarbeitet.

RND: Erzeugt eine Zufallszahl zwischen 0 und 1.

RUN: Start der Programmausführung.

STOP: Programmunterbrechung - Fortsetzung durch CONT.

3.11 Ausblicke

Sie haben jetzt einen kleinen Einblick in die Möglichkeiten des ST-BASIC bekommen. Für kleinere Programme reicht BASIC völlig aus, aber bei größeren Programmprojekten werden Sie den Editor wohl für unpraktisch halten, die Programme für nicht übersichtlich genug und den Programmablauf für zu langsam.

Schon seit längerer Zeit werden von Software-Firmen BASIC-Versionen vertrieben, die dem ST-BASIC in vielen Dingen

überlegen sind. Wir wollen Ihnen hier die zwei bekanntesten Versionen und ihre Vorzüge kurz vorstellen.

Das wohl bekannteste BASIC für den ST ist das GfA-BASIC von der Firma GfA Systemtechnik in Düsseldorf. Aufgrund der allgemeinen Verbreitung bekommt man zu diesem BASIC eine Reihe guter Bücher sowohl für den Einsteiger als auch für den fortgeschrittenen Anwender. Genannt werden sollen hier stellvertretend "Das große GfA-BASIC-Buch", "GfA Tips und Tricks" und ein übersichtliches Nachschlagewerk, der "GFA-BASIC-Führer", alle erschienen im Data Becker Verlag.

Das GfA-BASIC zeichnet sich durch einen guten und schnellen Editor aus. Programme werden in GfA-BASIC ohne Zeilennummern geschrieben und benutzen sogenannte "Labels" für goto-Befehle. Das Programmlisting wird sofort strukturiert und ist auch bedingt durch nur einen BASIC-Befehl pro Zeile sehr übersichtlich. Die sehr umfangreichen BASIC-Befehle sind dazu noch viel schneller als ST-BASIC. Durch einen sogenannten Compiler von GfA werden die Programme dann nochmals schneller und lassen sich ganz ohne BASIC als normales Programm mit Doppelklick starten.

Das zweite BASIC mit fast gleichen Möglichkeiten ist das Omikron BASIC. Es ist teilweise noch schneller als GfA-BASIC und legt großen Wert auf Rechengenauigkeit. Der Editor ist etwas gewöhnungsbedürftiger, läßt aber die Programmierung mit und ohne Zeilennummern zu. Auch dieses Programm wird durch einen Compiler noch schneller. Dieses BASIC ist eine direkte Konkurrenz zu GfA-BASIC, ist aber noch nicht so verbreitet. Auch Bücher zu Omikron BASIC sind noch rar.

Außer BASIC gibt es für den ST noch weitere interessante Programmiersprachen. So sind einige gute Compiler für die Sprachen C und Pascal auf dem Markt, die sehr leistungsfähig und mittlerweile ausgereift sind. Mit geeigneten Programmen können Sie den ST auch in Maschinensprache programmieren. Bei fast allen Programmen ist natürlich ein gutes Lehrbuch die Voraussetzung zum Erfolg.

4. In einer ruhigen Minute...

Jetzt haben wir zusammen den ATARI ST kennengelernt. Bewußt verzichteten wir bisher darauf, Sie mit komplizierten und im Moment vielleicht noch uninteressanten Hintergrundinformationen zu langweilen.

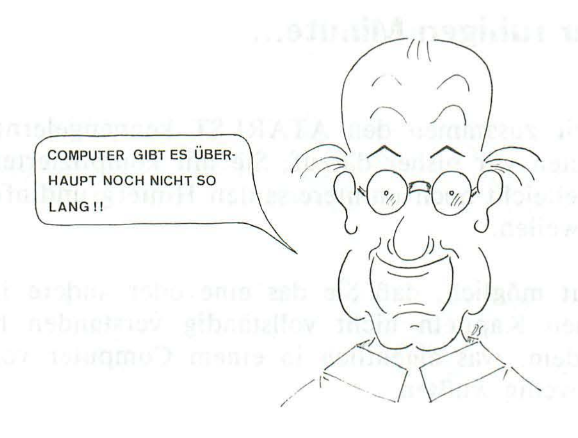
Es ist aber gut möglich, daß Sie das eine oder andere in den vorangegangenen Kapiteln nicht vollständig verstanden haben, weil Sie von dem, was eigentlich in einem Computer vor sich geht, noch zu wenig wußten.

Außerdem interessiert es Sie vielleicht, wie es zu der Entwicklung dieser Supermaschine ATARI ST kam, und was Sie in Zukunft noch alles damit anstellen können. All dies wollen wir nun in diesem Kapitel beleuchten. Am besten, Sie lesen dieses letzte Kapitel, wenn Sie einmal so richtig Zeit und Muße dazu haben. Interessant ist es sicherlich, was sich da noch so alles um Ihren ATARI ST herum getan hat und ständig tut.

4.1 Das Computer-Fachchinesisch

4.1.1 Wie schnell ist die Zeit vergangen...

Ein Besuch des Deutschen Museums in München lohnt allemal, nur sollte man dazu genügend Zeit mitbringen. In der Abteilung, die die Entwicklung des Computers eindrucksvoll vorführt, stößt man in einer Ecke auf eine große Rechenmaschine, die der Deutsche Konrad Zuse 1941 gebaut hat. Dies war der erste frei programmierbare Computer. Die Ausmaße entsprechen in etwa denen einer großen Schrankwand, die Leistungsfähigeit allerdings war im Vergleich zu heutigen Rechnern sehr gering.



Worauf ist einerseits die geringe Leistungsfähigkeit, andererseits die enorme Größe zurückzuführen? Eigentlich beide Male auf den gleichen Ursprung: die mechanische Art und Weise, mit der dieser Rechner an seine Arbeit geht.

Wir werden in diesem Kapitel noch einen kurzen Einblick in die Computerrechenweise bekommen, hier wollen wir uns mehr auf die Mechanik konzentrieren: Der Zuse-Rechner arbeitete mit einer Vielzahl von elektromechanischen Bauteilen, die, in einer Kette zusammengeschaltet, Informationen durch zahlreiche An- und Aus-Zustände festhalten konnten.

All das, was heute auf elektronischem Wege im Computer vor sich geht, geschah im Zuse-Rechner auf elektromechanische Weise. Ein ständiges Klappern der elektromechanischen Schalter (= Relais) lag über diesem Monstrum. Man kann sich leicht vorstellen, daß bei derartig vielen Schaltern der Verschleiß entsprechend hoch war. So war die Fehlerquote bei Rechnungen natürlich auch weitaus höher als es heute der Fall ist, wo Computer in der Regel (fast) fehlerfrei arbeiten. Auf einmal stimmte das durch Lämpchen angezeigte Rechenergebnis nicht mehr.

Auch die Wanzen fühlten sich in diesem Riesenrechner wohl und krochen hinein, was oftmals dazu führte, daß die Schalter innen nicht mehr richtig funktionierten und der Rechner falsche Ergebnisse lieferte. Aus dieser Zeit stammt übrigens auch der Ausdruck "Debuggen" (bug = Wanze; debuggen = von Wanzen befreien) für "Fehler suchen" bzw. "bug" für "Fehler".



Doch bleiben wir noch ein wenig bei der Computergeschichte. Wir haben festgestellt, daß der Zuse-Rechner zu langsam und vor allem zu groß war. Einige Jahre später hatten dann die Entwickler die rettende Idee: Aus dem so verschleißfreudigen Relais wurde der Transistor, ein erheblich kleinerer elektronischer Baustein, der zwar die gleiche Arbeit wie das Relais leistete, jedoch nicht mehr mechanisch arbeitete, sondern rein elektronisch (für die Erzeugung des An- bzw. Auszustandes zeichnet eins seiner drei Beinchen verantwortlich).

Ein Computer mit Speicherstellen in Form von Transistoren wurde schnell entwickelt, und der Größenunterschied zum Zuse-Rechner war schon gewaltig.

Doch die Erfinder und Entwickler arbeiteten weiter. Schließlich kam man auf die Idee, lediglich elektrische Bahnen, die die Transistoren in sich und untereinander miteinander verbinden, zweidimensional auf ein Papier aufzuzeichnen. Bei einer Anzahl von meist tausenden von Transistoren kann man sich gut vorstellen, wie groß diese Zeichnung aussehen mußte - also was den Miniaturisierungsfortschritt anbetrifft eigentlich ein Rückschritt.

Doch man hatte eine gute Idee: Die Riesenzeichnung der Verbindungen wurde abfotografiert. Bei einem entsprechend guten Filmmaterial hatte man nun mit einem Schlag aus einer mehrere Quadratmeter großen Zeichnung ein Abbild auf einigen Quadratzentimetern geschaffen.

Durch ein besonderes chemisches Verfahren wird die abfotografierte Strichzeichnung der Transistoren stromleitend gemacht ... und schon hat unser nun sogenannter "Chip" die Leistung von vielen tausend Transistoren. Nicht ganz: Erst muß unser Chip ja noch in Kontakt zu seiner Außenwelt treten: Dafür werden im Vergleich zum winzigen internen Leiternetz wie riesige Baumstämme erscheinende Leiterbahnen rund um den eigentlichen Chip aufgebracht, so daß nun jeder Mensch mit seinem Lötkolben die Verbindung nach draußen hin - vielleicht zu einem anderen Chip - selbst herstellen kann.

Im Vergleich zu Relais und Transistoren ist unser Chip zudem erheblich schneller geworden - bis zu 100000 Additionen können pro Sekunde durchgeführt werden -, denn das engverzweigte Leitungsnetz bietet erheblich kürzere Entfernungen als in der Größenausdehnung einer Schrankwand wie beim Zuse-Rechner.

Wie weit ist nun die Entwicklung dieser Chips bis heute vorangeschritten? In der Entwicklung sind gerade die sogenannten Megabitchips, d.h. Mega = 1 Million Transistorfunktionen auf der Fläche eines Stecknadelkopfes. Dieser vorerst letzte Schrei der Entwicklung befindet sich zwar bis jetzt noch nicht in Ihrem ST-Rechner, aber immerhin eine Vielzahl von 256 Kilo-

bit-Chips, die sage und schreibe schon 256000 Transistorfunktionen auf der Fläche eines Stecknadelkopfes zusammenfassen.

4.1.2 Verschiedene Chips

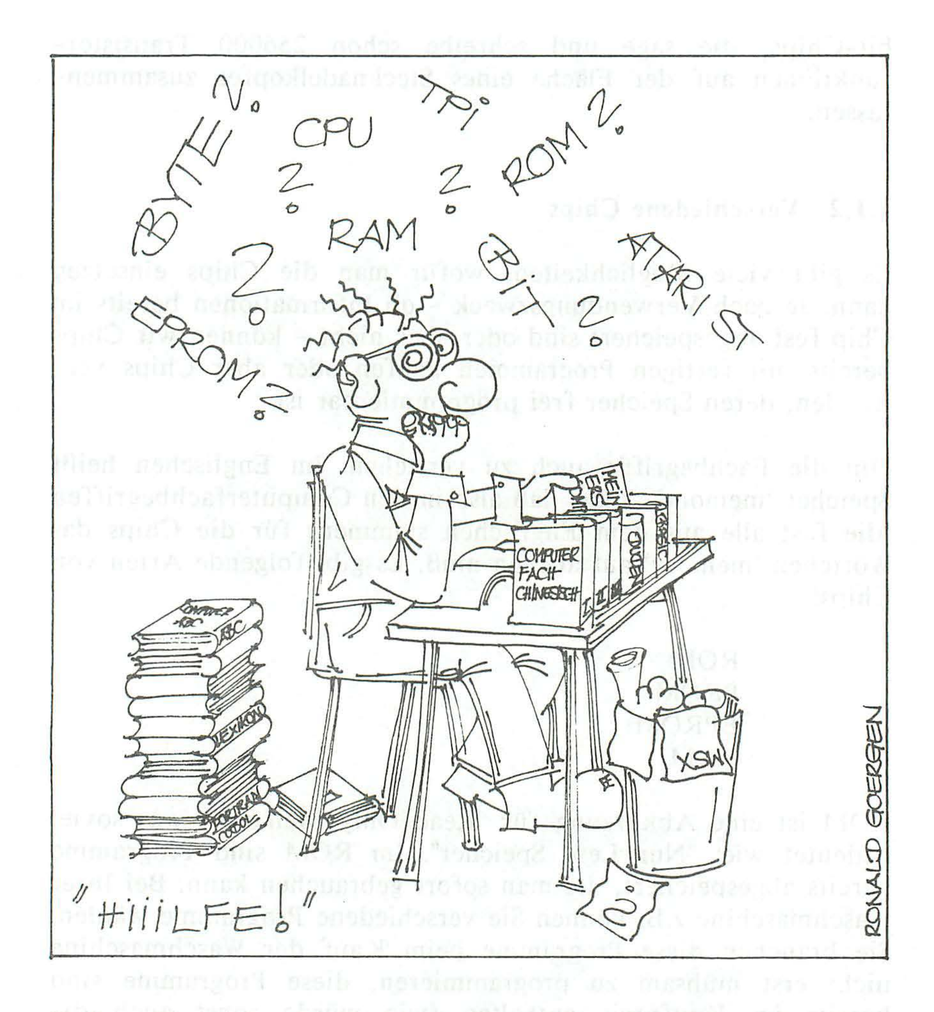
Es gibt viele Möglichkeiten, wofür man die Chips einsetzen kann. Je nach Verwendungszweck - ob Informationen bereits im Chip fest abgespeichert sind oder aber nicht - können wir Chips bereits mit fertigen Programmen kaufen oder aber Chips verwenden, deren Speicher frei programmierbar ist.

Um die Fachbegriffe auch zu verstehen: Im Englischen heißt Speicher 'memory'. Klar, daß also in den Computerfachbegriffen (die fast alle aus dem Englischen stammen) für die Chips das Wörtchen 'memory' auftauchen muß. Es gibt folgende Arten von Chips:

ROM
PROM
EPROM
RAM

ROM ist eine Abkürzung für "Read Only Memory", was soviel bedeutet wie: "Nur Lese Speicher". Im ROM sind Programme bereits abgespeichert, die man sofort gebrauchen kann. Bei Ihrer Waschmaschine z.B. können Sie verschiedene Programme wählen. Sie brauchen diese Programme beim Kauf der Waschmaschine nicht erst mühsam zu programmieren, diese Programme sind bereits im Kaufpreis enthalten (wie würde sonst auch die Wäsche einer Hausfrau aussehen, die sich vorher noch nicht mit der Programmierung beschäftigt hat?!).

In Ihrem ST haben Sie immerhin das Betriebssystem schon im ROM. Programmiersprachen und Anwendungsprogramme aber müssen von Diskette geladen werden.



Stellen Sie sich ein ROM so vor, wie wir vorhin die Entwicklung des Chips beschrieben haben. Allerdings sind hier die Transistoren bzw. die einzelnen Leiterbahnen bereits in ihrem Zusammenspiel von Entwicklern bzw. Programmierern einmal vorbelegt worden. So hat man beim GEM- und TOS-ROM also nicht die Möglichkeit, diese Leiterbahnen neu zu beeinflussen, Sie könnten ihren Inhalt auslesen und gebrauchen, aber nicht verändern.

In der Vorphase zur Entwicklung von Computern werden nicht sofort ROMs verwendet (dann wären die Entwicklungskosten noch weitaus höher durch das andauernde Vernichten der im Vorstadium nicht richtig programmierten ROMs).

Man verwendet deshalb *PROMs* (Programmable ROMs). Das PROM ist ähnlich dem EPROM, auf das wir gleich zu sprechen kommen, nur das 'E' (für "Erasable" = löschbar) fehlt: Ist ein PROM einmal programmiert, muß man sich mit dem Inhalt darauf zufriedengeben. Um den Inhalt zu verändern, muß ein neues PROM programmiert werden.

Außerdem gibt es noch *EPROMs* (Erasable Programmable ROM = löschbares, programmierbares ROM). Hiermit ist es dem Anwender möglich, die Struktur der Leiterbahnen selbst festzulegen, also ein EPROM zu "brennen" und bei Nichtgefallen unter UV-Licht die Informationen wieder zu löschen (Englisch: to erase). Wörtlich übersetzt ins Deutsche würde EPROM also eigentlich heißen: Löschbarer und programmierbarer Nur-Lese-Speicher.

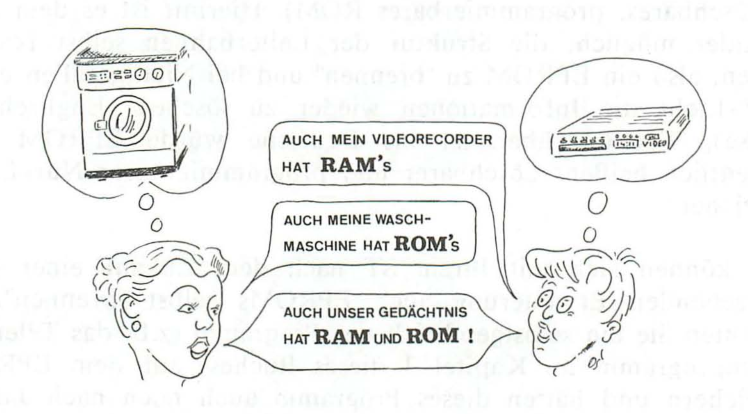
Sie können sich mit Ihrem ST nach dem Zukauf einer entsprechenden Erweiterung auch EPROMs selbst "brennen". So könnten Sie ein selbstgeschriebenes Programm (z.B. das Telefonbuchprogramm im Kapitel 3 dieses Buches) auf dem EPROM speichern und hätten dieses Programm auch noch nach Jahren ohne neues Laden von der Diskette oder der Harddisk augenblicklich nach dem Einschalten des Computers verfügbar.

Noch etwas zum Aussehen der EPROMs: EPROMs erkennen Sie gut daran, daß auf ihre Oberseite meist kleine Zettelchen aufgeklebt sind. Wenn Sie so einen Chip im Bastlerladen sehen, lösen Sie das Zettelchen ruhig kurz einmal ab: darunter sehen Sie dann den eigentlichen Chip, wie eine kleine Maus im großen Käfig. Die Größe des Chips wird durch die vorhin schon angesprochenen Leiterbahnen nach draußen hin bestimmt.

Warum nun dieses Zettelchen oben auf dem EPROM? Sollen hierdurch gar Geheimnisse geschützt werden, die Sie nun entschlüsselt haben? Nein, wir haben doch bereits erfahren, daß das

Programm auf dem EPROM nur durch UV-Licht wieder gelöscht werden kann. Wie könnte sonst UV-Licht auf den Chip treffen, wenn nicht durch das kleine Sichtfenster nach Abnehmen des Zettelchens.

Die wichtigsten Chips im Computer sind die *RAMs*, die sich alle eingegebenen Datenmengen merken können und sie auf Befehl oder durch Stromausfall auch sofort wieder vollständig vergessen. Denken Sie an Ihren Videorecorder, bei dem Sie alle paar Tage neue An- und Ausschalt-Zeitpunkte einprogrammieren können. Auch dort sind freiprogrammierbare Speicher = die RAMs für diesen Zweck enthalten.



RAM heißt aber nun nicht 'Read A Memory', sondern 'Random Access Memory', frei übersetzt: Speicher mit direkter Zugriffsmöglichkeit. Zwar können wir auch beim ROM direkt auf die einzelnen Speicherstellen, die Transistoren, zugreifen, aber wir können diese nicht wie beim RAM für unsere höchstpersönlichen Zwecke verändern.

Also ist das RAM unser eigentlich frei programmierbarer Speicher: hier können wir unsere Daten, Programmierzeilen oder auch Liebesbriefe ablegen. Jedoch Vorsicht: Wollen wir diese Daten für immer behalten, müssen wir den Computer und damit seine RAMs unter Strom lassen oder aber die dort enthaltene

Information auf andere magnetische Speichermedien wie Diskette oder Harddisk übertragen. Ähnlich ist es ja auch, wenn Sie Musik hören: Sie vergessen so manche Tonfolge, wenn Sie die Musik nur einmal hören; zeichnen Sie diese allerdings auf Tonband auf, ist die Erinnerung immer wieder abrufbar.

4.1.3 Eigentlich sind Computer dumm

Sie erinnern sich noch an die Funktionsweise des Zuse-Rechners? Viele Relais (Schalter), die nur die Zustände "Strom fließt" und "Strom fließt nicht" wiedergeben konnten, waren jeweils aneinandergereiht, und so zeigten kleine Lämpchen schließlich nach der Berechnung das Ergebnis an.

Wie arbeitet nun ein Computer - nicht nur unser ST - allein mit diesen beiden Zuständen 'an' und 'aus'? In grauer Vorzeit haben sich unsere Vorfahren zusammengesetzt und willkürlich ein Zahlsystem entwickelt, das zehn verschiedene Ziffern kennt:

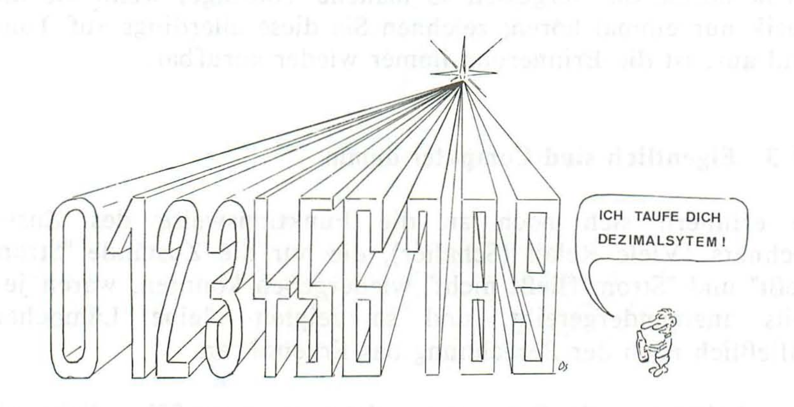
0 1 2 3 4 5 6 7 8 9

Wenn man zählt, fängt man also ganz links in der obigen Reihe bei 0 an. Nach kurzer Zeit kommt man bei der 9 an. Was jetzt? Richtig! Man beginnt wieder ganz vorne, stellt aber davor eine 1 und hat die Zahl 10. Die Stelle "1" bleibt nun gleich, bis man mit der zweiten Stelle wieder bei der 9 angekommen ist, und wieder bei "0" anfängt, nur statt der "1" nun eine "2" als erste Stelle hat, und somit die Zahl 20 erhält. So geht es immer weiter.

Wenn man bei beiden Stellen bei "9" (=99) angekommen ist, werden beide Stellen "0" und man stellt wieder eine "1" davor. Die Zahl "100" ist erreicht. So geschieht dies bei Hundertern, Tausendern usw.

Obgleich wir Menschen uns schnell daran gewöhnt haben, diese 10 Ziffern richtig und logisch in einem Zahlsystem, dem sogenannten Zehner- oder Dezimalsystem, zu gebrauchen, müssen

wir doch immer daran denken, daß dies ein Zahlensystem ist, das völlig willkürlich ausgesucht wurde.



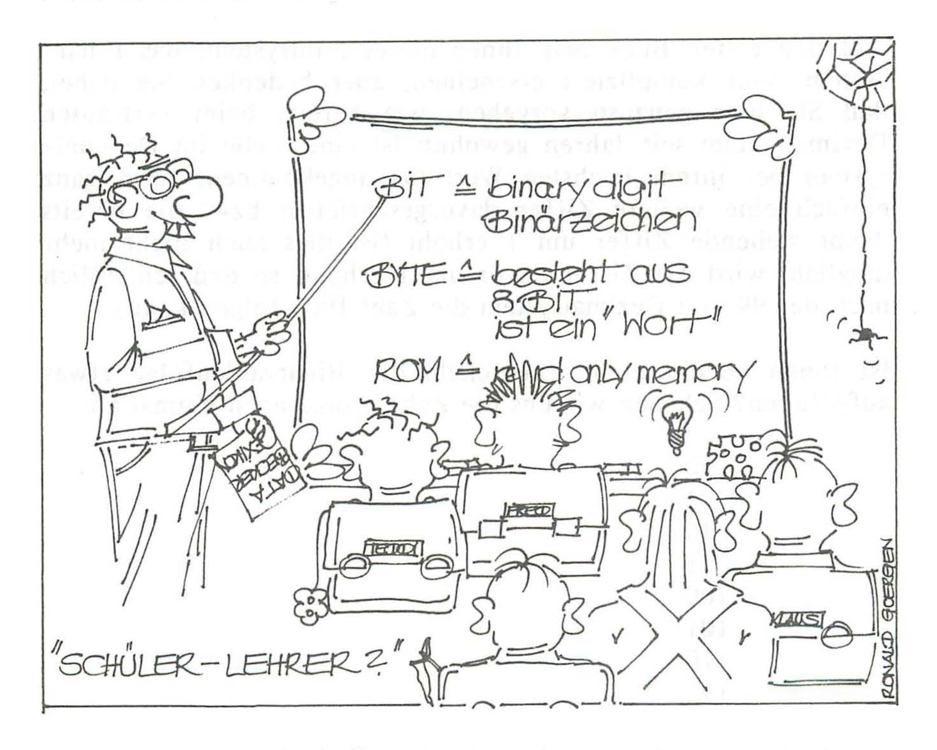
Als sich die Menschheit nun daran machte, Computer zu erfinden, mußte man darüber nachdenken, wie dem Computer auch ein Zahlsystem, am besten unser Dezimalsystem, beigebracht werden könnte. Schließlich konnte man dem Computer nicht einfach das Dezimalsystem beibringen, wie man es selbst gelernt hatte.

Es gab Entwicklungen, die mit Schaltern arbeiteten, die mehr als zwei Zustände unterscheiden konnten (es fließt viel Strom, es fließt wenig Strom, es fließt kein Strom). Jedoch, und das war doch die eigentliche Grundidee zur Entwicklung der Computer, manchmal erkannte der Rechner nicht den Unterschied z.B. zwischen wenig und kein Strom. Ergebnis: die Rechnung stimmte nicht mehr, das Resultat war falsch (unvorstellbar, wenn unsere Taschenrechner heute so unsicher rechnen würden).

Also einigte man sich schließlich darauf, den Computer mit nur zwei Zuständen arbeiten zu lassen: an und aus. Hierbei konnte man allerdings dann zu 100 Prozent sichergehen, daß dieses System auch einwandfrei funktionierte! Ein Name für dieses Rechensystem, das nur zwei Zustände kennt, nämlich 'aus' und

'an' oder 0 und 1, war bereits zuvor von Mathematikern bei der Entwicklung unseres Dezimalsystems geprägt worden: Binär- oder Dualsystem.

Wie ist es nun möglich, nur mit den Ziffern 0 und 1 auch größere Zahlwerte zu verarbeiten? Man könnte doch einfach, entsprechend der Größe der Zahl, Einsen benutzen, also z.B. bei der Zahl 10 zehn Einsen und bei der Zahl 300 gar dreihundert Einsen. Sicherlich wäre das eine Möglichkeit, aber in diesem Fall würden wir ja nur eine Ziffer, die "1" benutzen und brauchten die"0" nicht.



Gehen wir doch einfach mal das Binärsystem so durch, wie wir dies vorhin bereits beim Dezimalsystem Schritt für Schritt getan haben: Wir haben zwei Ziffern zur Verfügung, die wir allerdings bei zunehmender Größe der Zahl miteinander verknüpfen können. Die kleinste Binärzahl ist die 0, dann folgt die 1. Da wir nicht wie beim Dezimalsystem über eine größere Ziffer als

die 1 verfügen, müssen wir nun schon eine Ziffer voranstellen - also die nächstgrößere Zahl ist die 10 (bitte lesen Sie beim Arbeiten im Binärsystem immer Ziffer für Ziffer - also in diesem Fall nicht 'Zehn' sondern 'Eins-Null').

Danach können wir die kleinste Stelle wieder um 1 vergrößern und es entsteht so die Binärzahl 11. Wieder sind die dargestellten Ziffern bei ihrem Höchstwert angelangt (ähnlich wie bei unserem Dezimalsystem die Zahl 99) und wir müssen erneut eine Ziffer davorsetzen: 100. Daß die weiteren Werte 101, 110 und 111 sind, können Sie sich sicher nun selber errechnen.

Auf den ersten Blick mag Ihnen dieses Zahlsystem, das Binärsystem, sehr kompliziert erscheinen, aber bedenken Sie dabei, daß Sie hier genauso vorgehen, wie bereits beim vertrauten Dezimalsystem seit Jahren gewohnt: Ist eine Stelle im Dezimalsystem bei ihrem höchsten Wert (9) angekommen, wird ganz einfach eine weitere Ziffer davorgeschrieben bzw. die bereits davor stehende Ziffer um 1 erhöht (ist dies auch nicht mehr möglich, wird die Ziffer davor um 1 erhöht, so daß schließlich nach der 999 im Dezimalsystem die Zahl 1000 folgen kann).

Ist Ihnen vorhin beim Entwickeln der Binärzahlenfolge etwas aufgefallen? Schauen wir uns die Zahlenfolge noch einmal an:

```
  0
  1
 10
 11
100
101
110
111
```

Wir hatten uns doch vorhin darüber Gedanken gemacht, wie man größere Zahlen aus dem Dezimalsystem in das Binärsystem übertragen kann. Erst hatten wir angenommen, für jede Zahl die entsprechende Anzahl von Einsen nehmen zu müssen, also z.B. für die Zahl 10 auch zehn Einsen. Schauen wir aber nun unsere Folge von der 0 bis zur 111 an, können wir feststellen, daß wir

zwischendurch nicht nur die Zustände 1 und 11 errechnet haben, sondern etliche mehr (10, 100, 101 und 110). Nun wollen wir den Wert unserer Binärzahlenfolge dadurch deutlicher machen, indem wir die entsprechende Zahl des uns bereits mehr vertrauten Dezimalsystems in aufsteigender Reihenfolge danebenschreiben:

Dezimal	Binär
0	0
1	1
2	10
3	11
4	100
5	101
6	110
7	111

Also brauchen wir für die 7 nicht, wie vorhin angenommen, sieben Einsen, sondern lediglich drei. Sie werden sich nun aber fragen, wie man denn dieses System auch in die Zehner und Hunderter weiterführen kann ... das würde ja Ewigkeiten dauern, wenn man immer weiter hochzählen müßte. Keine Angst. Diesem logischen System liegt selbstverständlich auch ein mathematisches System zugrunde. Wir benötigen dazu die verschiedenen Potenzen der Zahl 2:

$2^0 = 1$
$2^1 = 2$
$2^2 = 4$
$2^3 = 8$
$2^4 = 16$
$2^5 = 32$
$2^6 = 64$
$2^7 = 128$

So vereinbarte man, jede Binärzahl in folgende Tabelle einzuordnen.

...	2^5 =32	2^4 =16	2^3 =8	2^2 =4	2^1 =2	2^0 =1	
			1	1	0	0	
=			8	+ 4			=12

Nun fängt man an zu addieren. Immer da, wo eine "0" in der Binärzahl steht, kann man weiterlesen, da wo eine "1" zu finden ist, addiert man die Zweierpotenz (die Zahl in der zweiten Zeile). Bei der Binärzahl "1100" ergibt dies eben 12, weil (von hinten gesehen) nur die dritte und vierte Stelle eine "1" enthalten, und die addierten Zweierpotenzen 12 betragen.

Ein weiteres Beispiel:

Binär 111 = Dezimal $2^2 + 2^1 + 2^0 = 4 + 2 + 1 = 7$

Vielleicht können Sie sich somit auch vorstellen, daß man mit lediglich acht Binärziffern bereits die Dezimalzahl 255 erzeugen kann, bei sechzehn Binärstellen sogar die Dezimalzahl 65535.

Genug mit diesen schrecklichen Mathe-Spielen. Es wäre schön, wenn Sie nun einen kleinen Einblick in diese 0-1-Rechenweise Ihres Computers gewonnen haben, denn in der weiteren Arbeit mit Ihrem ST werden Sie immer wieder auf dieses Zahlsystem gestoßen werden. Dies ist nicht nur bei Ihrem ST-Rechner so, auch bei allen anderen Maschinen oder Geräten auf dem Markt, die irgendwo einen Computer eingebaut haben. Nur mit diesem Zahlsystem ist es möglich, einem nicht zum Denken fähigen Apparat zumindest den Vorgang des Rechnens beibringen zu können. Da aber alle Computerlogik auf Rechenvorgängen beruht, ist durch die mathematisch-logische Verknüpfung im Binärsystem das eigentliche Fundament für einen Computer geschaffen worden.

Bevor wir wieder in die Praxis eintauchen, noch ein wichtiger Hinweis: Neben dem Binärsystem und dem Dezimalsystem gibt es noch viele andere Zahlsysteme: Zahlsysteme, in denen es acht verschiedene Zustände gibt:

0 1 2 3 4 5 6 7 = das Oktalsystem

oder gar sechzehn verschiedene Zustände (0 bis 9 und dann fängt man einfach mit dem ABC an), also

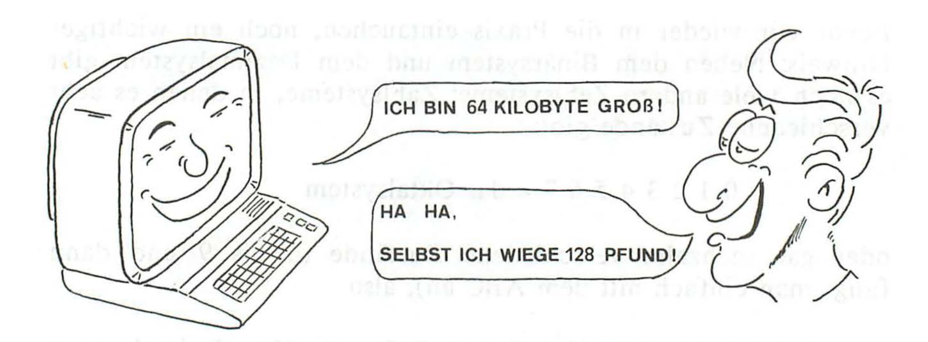
0 1 2 3 4 5 6 7 8 9 A B C D E F = das Hexadezimalsystem

Die Logik dieser Systeme funktioniert genauso, wie wir sie gerade auch beim Dezimal- und beim Binärsystem kennengelernt haben.

Übrigens: Im Anhang dieses Buches finden Sie kleine BASIC-Programme, mit denen Sie diese Umrechnungen in die verschiedenen Zahlsysteme leicht von Ihrem ST erledigen lassen können.

4.1.4 Was heißt 512 Kilobyte?

Sie erinnern sich noch daran, was RAM ist? Richtig! RAM ist ein Speicher in Ihrem Computer, in dem Sie später Ihre Programme, Texte usw. ablegen können. Wenn Sie Ihren ST ausschalten oder wenn der Strom ausfällt, ist sämtliche Information aus dem RAM-Speicher verschwunden, ähnlich wie beim RAM im Videorecorder: Fällt der Strom aus, blinkt die Anzeige im Videorecorder wie wild, um Sie darauf aufmerksam zu machen, daß alle Programminformationen und meist sogar die vorher eingestellte Uhrzeit vergessen wurden (wie bei einigen Videorecordern gibt es allerdings neuerdings auch bei einigen Computern eine sogenannte Batteriepufferung für den RAM-Speicher, d.h. nach dem Wegfall des Stromes holt sich der Computer den Strom für die RAMs aus einer Batterie.



Die ATARI STs haben einen frei programmierbaren Speicher von 512 KByte aufwärts. Auch andere Computer werden qualitativ mit solchen Zahlen eingeordnet, ähnlich wie bei Autos mit der PS-Zahl. Was bedeutet nun die Abkürzung KByte oder K?

Jede Stelle in einer Binärzahl nennt man ein "Bit" (Abk. f. BInary DigiT= Binäre Ziffer). Ist ein Bit (als eine Ziffer in der Binärzahl) 1, sagt man, das Bit ist "gesetzt". Ist die Ziffer "0", spricht man von einem "nicht gesetzten" Bit. Acht Bits bilden das sogenannte "Byte". Mit acht binären Ziffern kann man maximal die Zahl 255 darstellen.

Weil die gesamten Speicherbausteine auf diesem Prinzip beruhen, kommen immer fürchterlich krumme Zahlen dabei heraus, wenn man den gesamten Speicherplatz zusammenrechnet. Man dachte sich daher eine Maßeinheit aus, das Kilobyte (Kilo heißt eigentlich 1000), das 1024 Bytes beinhaltet, und das Kilo-Bit, das 1024 Bits umfaßt.

So gibt es nun z.B. Chips, die 256 KiloBits speichern können. Solche Chips sind übrigens auch in Ihrem ST zu finden. Umgerechnet in Bytes (1 Byte = 8 Bit) ergeben 256 KBit = 32768 Bytes, die dann als 32 KByte bezeichnet werden. 16 dieser 32 KByte-Chips findet man z.B. im ATARI 260 ST und im ATARI 520 ST/M, so daß sich ein Speicherplatz von 16 x 32 KByte = 512 KByte ergibt. 32 dieser Chips erfüllen Ihren Dienst im 1040 ST, der es damit auf 1 MByte (ca. 1024 KByte) bringt.

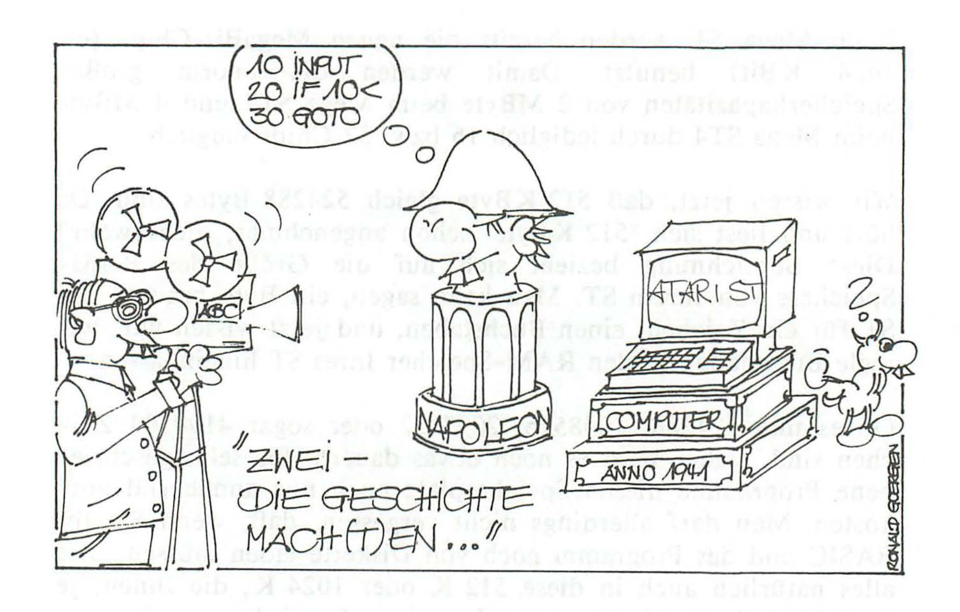
Beim Mega ST werden bereits die neuen MegaBit-Chips (ca. 1024 KBit) benutzt. Damit werden die enorm großen Speicherkapazitäten von 2 MByte beim Mega ST2 und 4 MByte beim Mega ST4 durch lediglich 16 bzw. 32 Chips möglich.

Wir wissen jetzt, daß 512 KByte gleich 524288 Bytes sind. Da hört und liest sich "512 KByte" schon angenehmer, nicht wahr? Diese Bezeichnung bezieht sich auf die Größe des RAM-Speichers von Ihrem ST. Man kann sagen, ein Byte braucht der ST für ein Zeichen, einen Buchstaben, und jetzt wissen wir, wie viele Buchstaben in den RAM-Speicher Ihres ST hineinpassen.

Ob es nun 524288, 1048576, 2097152 oder sogar 4194304 Zeichen sind, sicher wird es noch etwas dauern, bis selbstgeschriebene Programme diesen Speicherplatz auch nur annähernd auskosten. Man darf allerdings nicht vergessen, daß, wenn Sie Ihr BASIC und das Programm noch von Diskette laden müssen, dies alles natürlich auch in diese 512 K oder 1024 K, die Ihnen, je nach Modell, zur Verfügung stehen, gepackt wird.

Vielleicht sind Sie aber auch in der glücklichen Lage, schon einen ST zu besitzen, bei dem BASIC fest eingebaut auf ROMs sitzt, und Sie die RAMs in ihrer vollen Größe frei benutzen können. Sehr positiv wirkt sich viel Speicherplatz immer dann aus, wenn man große Textmengen zu bewältigen hat. Dieses Buch besteht z.B. aus ca. 500000 Zeichen, die wir alle gleichzeitig im Speicher unseres ATARI ST halten könnten, weil der Speicherplatz eben so schön groß ist. Ein enormes Preis-/ Leistungsverhältnis, nicht wahr!?

Was heißt das nun, 524288 oder 1048576 Buchstaben? Auf jeder Seite dieses Buches sind ungefähr 2000 Buchstaben abgedruckt. Sie können sich nun selbst ausrechnen, daß bei 512 Kilobyte oder einem Speichervermögen von 524288 Buchstaben ca. 260 Buchseiten mit Text in Ihren ATARI ST-Rechner hineinpassen würden. Mit einem 4 MByte großen Speicher könnten Sie dann sogar über 2000 Seiten bearbeiten. Fast unvorstellbar, nicht wahr?



4.2 Zukunftsaussichten

Jetzt wissen Sie schon eine ganze Menge über Ihren ATARI ST. Sicher haben Sie schon darüber nachgedacht, wie Sie Ihre Computeranlage noch erweitern und verbessern können.

Der ATARI ST verfügt über eine Vielzahl von Anschlüssen, so daß es möglich ist, fast alles, was es derzeitig auf dem Markt gibt, auch an ihn anzuschließen. Außerdem gibt es bereits viele fertige Programme für den ST, mit denen Sie Textverarbeitung, Dateiverwaltung, Spiele und und und betreiben können. Wir wollen Ihnen jetzt einen kleinen Überblick darüber geben, was Sie noch so alles mit Ihrem ST machen können.

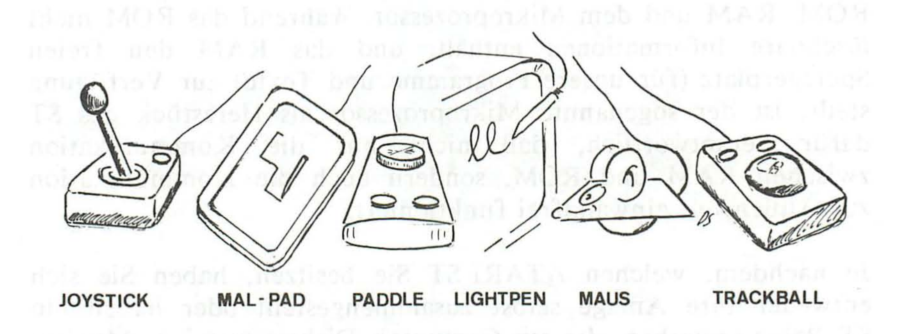
4.2.1 Anschlußfreudig

Wie wir jetzt bereits wissen, besteht ein Computer u.a. aus ROM, RAM und dem Mikroprozessor. Während das ROM nicht löschbare Informationen enthält, und das RAM den freien Speicherplatz (für unsere Programme und Texte) zur Verfügung stellt, ist der sogenannte Mikroprozessor als Herzstück des ST dafür verantwortlich, daß nicht nur die Kommunikation zwischen RAM und ROM, sondern auch die Kommunikation zur Außenwelt einwandfrei funktioniert.

Je nachdem, welchen ATARI ST Sie besitzen, haben Sie sich entweder Ihre Anlage selbst zusammengestellt oder haben ein ST-Paket erworben, das aus Computer, Diskettenstation, Monitor und Maus besteht.

Außer diesen mindestens für eine ST-Anlage erforderlichen Einzelteilen können Sie aber noch eine Vielzahl von anderen Dingen an Ihren ATARI ST anschließen. Fangen Sie schon einmal an zu sparen, denn wenn Sie die folgenden Seiten gelesen haben, wird der Wunsch nach Vergrößerung Ihrer Anlage sicherlich da sein!

4.2.1.1 Zwei Joystickports



Joystick

Die Aktionen unseres ST können zum Teil mit Joysticks "ferngesteuert" werden. Zu diesem Zweck verfügen alle ATARI ST-Computer an der rechten Seite bzw. an der Unterseite über zwei Buchsen (eine Buchse davon benutzen wir bereits für die Maus).

Wir können an diese Buchsen nicht nur für zwei Spieler sogenannte Joysticks (Steuerhebel zum Spielen und Abfeuern) anschließen, auch andere Steuerinstrumente, wie Paddles und in Zukunft vielleicht sogar ein Zeichenbrett, sind dort jederzeit willkommen.

Die Bewegung des eigentlichen Joystickhebels wird pro Joystickport nicht nur in den vier Himmelsrichtungen Norden, Westen, Süden, Osten, sondern auch in den Zwischenhimmelsrichtungen (z.B. Nordost, Südwest ...) abgefragt (man könnte auch sagen, daß alle Gradstellungen zwischen 0 und 360 Grad im 45 Grad-Abstand gemessen werden, das sind insgesamt je Joystick 8 Positionen).

Durch das Andrücken des Joystickhebels in die gewünschte Richtung können Sie somit das auf dem Bildschirm befindliche Objekt in die gewünschte Richtung steuern. Es ist zwar eine Möglichkeit, den Joystick samt seinen Knöpfen für ordinäre Spiele zu gebrauchen, es ist aber selbstverständlich auch möglich, damit interessantere Dinge zu tun, z.B. ein Bild zu malen, es beim Druck auf den Joystickknopf auf Diskette abzuspeichern, oder es auf diese Art und Weise wieder in den Bildschirmspeicher einzuladen. Sie können sich selbst weitere Anwendungsmöglichkeiten für Joysticks überlegen, wozu Sie diese Steuergeräte an Ihren ST anschließen wollen.

Malpad

Das zuletzt genannte Beispiel (Zeichnen mit Joystick) kann viel einfacher und besser mit einem sogenannten Pad gelöst werden, das Sie ebenfalls an Ihren ST über die Joystickports ohne Mühe anschließen können. Beim Pad haben Sie eine etwa postkartengroße Malfläche zur Verfügung, die in ihrem verkleinerten Aufbau dem Bildschirm entspricht.

Zu dem Pad wird normalerweise ein Plastikgriffel geliefert. Diesen Malgriffel (im Notfall tut es auch der Fingernagel - aber Vorsicht vor Kratzern!) setzen Sie auf das Malpad auf, und nun können Sie auf dem Bildschirm die schönsten Gemälde malen, nur mit dem Unterschied zu einem gewöhnlichen Bild, daß Sie hier eigentlich nur den Griffel auf die Maltafel aufdrücken, die im gleichen Augenblick den exakten Aufdruckpunkt dem Computer weitervermittelt.

Sicherlich eine bessere Möglichkeit, seinem künstlerischen Drang ein Medium an die Hand zu geben, als mit Hilfe eines Joysticks (der nach dem Loslassen immer wieder in die Mittelposition zurückkehrt); allerdings ist ein Pad nicht nur erheblich teurer als ein Joystick (schließlich müssen alle Punkte auf dem Pad mit Sensoren genauestens abgefragt und die Information an Ihren ST-Rechner weitervermittelt werden), sondern auch nur zum Zeichnen gedacht.

Selbstverständlich verfügt das Pad auch über ein bis zwei Andruckknöpfe, die sich teils an der Außenseite des Pad, teils direkt am Plastikgriffel befinden. Zudem wird es sicherlich in Zukunft Software für die Pads geben, die eine Art Menü auf dem Bildschirm für die verschiedenen Funktionen ausgeben; Sie setzen den Plastikgriffel dann einfach dort auf das Pad, wo die von Ihnen gewünschte Funktion steht (z.B. Abspeichern oder Farbwahl ...).

Paddle

Artverwandt zum Joystick ist ein drittes Steuerinstrument, das Sie an einen oder beide Joystickports (letzteres bei zwei Mitspielern) anschließen können: das Paddle. Obgleich der Name ähnlich wie Pad klingt, haben diese zwei nichts miteinander zu tun. Das Paddle besteht aus einem Drehknopf, mit dem Sie einen Gegenstand je nach Drehrichtung nach links oder nach rechts steuern können.

Um sich die Funktion eines Paddles besser vorstellen zu können, hier zwei Beispiele, wie ein Programm ein Paddle benutzen kann:

1. Sie wollen ein Auto auf dem Bildschirm lenken. Dazu benutzen Sie den Drehknopf des Paddles wie ein Autolenkrad. Drücken Sie an der einen Seite des Paddles den auch hier befindlichen Knopf, fährt Ihr Wagen schneller, lassen Sie ihn wieder los, verlangsamt sich die Geschwindigkeit. Hat Ihr Paddle gar auch noch einen zweiten Knopf, der von Ihrem ST (Joystickport 0) vielleicht anders interpretiert wird, können Sie hiermit kuppeln.

2. Sie wollen ein Männchen in einem Spiel auf einer Wippe hochspringen lassen. Trifft das Männchen besonders günstig auf der Wippe auf, wird es entsprechend sehr hoch befördert und kann dort Luftballons abtreffen. Aufgrund der Schwerkraft kommt das Männchen schließlich wieder zum Erdboden her-

unter, und Sie müssen nun die Wippe so mit Ihrem Paddle neu justieren, daß das Männchen dort auch wieder günstig auftreffen kann.

Beide Spiele ließen sich zwar auch mit einem Joystick realisieren, wären dann aber bei weitem nicht so praktisch und - insbesondere wie beim Autorennen - nicht so wirklichkeitsgetreu zu bedienen.

Trackball

Hierbei handelt es sich um ein handtellergroßes Kästchen, das auf den Tisch gelegt wird. Der Name dieses Steuergeräts rührt von einer Kugel (Englisch: ball) her, die sich in dem Kästchen befindet. Sie ist so groß, daß sie nach draußen ragt und von einer darauf gelegten Hand nicht nur berührt, sondern auch mit Leichtigkeit in alle Richtungen bewegt werden kann. Die Funktion des Trackballs entspricht somit der eines Joysticks (8 Richtungen der Kugelbewegung werden registriert), wobei der Trackball allerdings sehr viel leichter zu handhaben ist.

Maus

Eine ähnliche Technik wie beim Trackball wird durch die an Ihren ST angeschlossene Maus erreicht. Dies ist eigentlich auch ein Trackball, nur die Kugel wird erst dadurch in Bewegung gesetzt, daß das Kästchen auf der Tischfläche irgendwo hinbewegt wird (die Kugel sitzt nämlich wie ein Rad an der Unterseite des Kästchens).

Es ergibt sich von selbst, daß sowohl Maus als auch Trackball über mindestens einen Steuerknopf verfügen, der die gleichen Funktionen auslösen kann, wie beim Joystick bereits ausführlich erläutert. Wird die ATARI-Maus an Joystickbuchse 0 angeschlossen, können die zwei Tasten auf ihrer Oberseite für zwei unterschiedliche Funktionen abgefragt werden.

Lightpen

Sie erinnern sich noch an das Malpad? Hierbei benutzen Sie einen Plastikgriffel und vollziehen Ihre kreative Tätigkeit auf einem postkartengroßen Plastikbrettchen mit Sensorzellen.

Der Lightpen ist auch ein Malwerkzeug, nur malen Sie hier nicht auf einer Zwischenstation zum Bildschirm wie dem Malpad, sondern direkt auf der Bildschirmoberfläche. Keine Angst, Ihr Monitorschirm wird hierbei weder verkratzt noch mit den verschiedensten Farben beschmiert. Wie der Name Lightpen (Lichtgriffel) bereits ausdrückt, sitzt in diesem Steuerinstrument eine Fotodiode, die genau feststellen kann, an welchem Bildschirmpunkt Sie gerade Ihre Künste schweifen lassen.

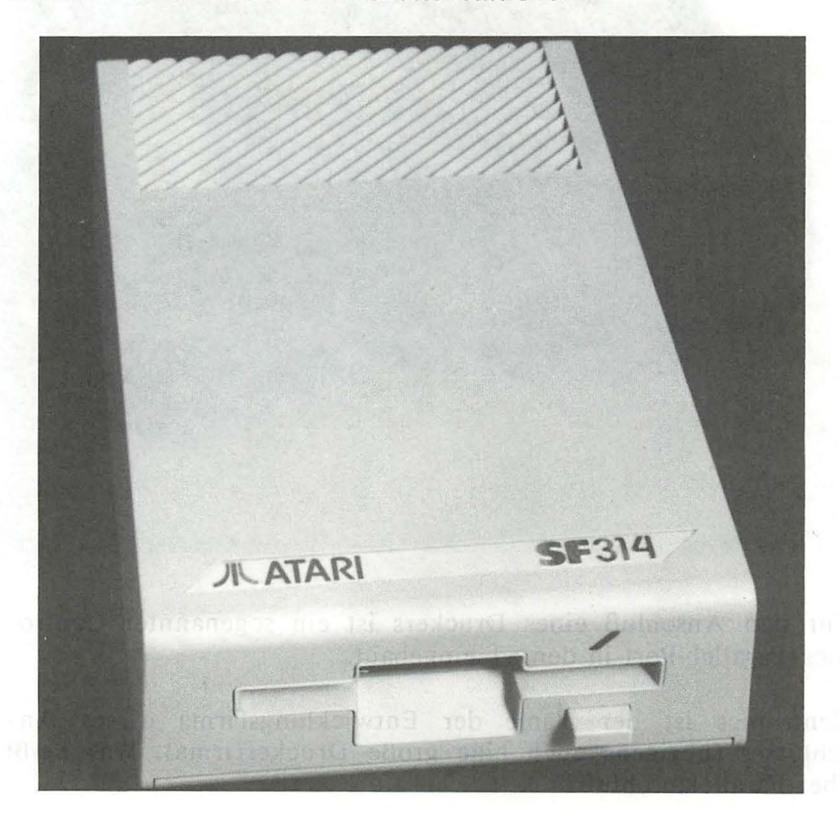
Während die Abfrage des Lightpen auf einen speziellen Computermonitor zufriedenstellend verläuft, sieht dies beim Anschluß an einen gewöhnlichen Fernseher schon ganz anders aus: Bei Lightpens auf dem Zubehörmarkt für andere Computer sind die Ergebnisse nicht in jedem Fall zufriedenstellend, da die dicke Glasscheibe unserer Fernsehgeräte die Lichtstrahlen nach außen hin derart ablenkt, daß der aufgesetzte Lightpen nicht dort malt, wo er eigentlich soll, sondern meist einige Zentimeter links oder rechts, oben oder unten vom eigentlich angesteuerten Bildschirmpunkt entfernt. Besser gelingt dies zwar mit einem Farbmonitor mit abgenommener Kontrastscheibe (falls vorhanden), jedoch zu mehr als einer Spielerei (oder einer Menüauswahl) ist dieses System im Augenblick kaum zu gebrauchen. Als Effekt jedoch immer sehr reizvoll.

Beim Schreiben dieser Zeilen war es noch nicht abzusehen, ob in nächster Zukunft ein Lightpen für die ATARI ST-Computer auf den Markt kommen wird. Außerdem stand noch nicht fest, ob der Lightpen am Joystickport angeschlossen wird.

Es gibt sicherlich noch viele andere Möglichkeiten (gerade für Bastler, die Steuerungen konrollieren wollen oder für Biologen, die die Messung eines Gewässers überwachen wollen), die Joystickbuchsen Ihres ST sinnvoll zu nutzen.

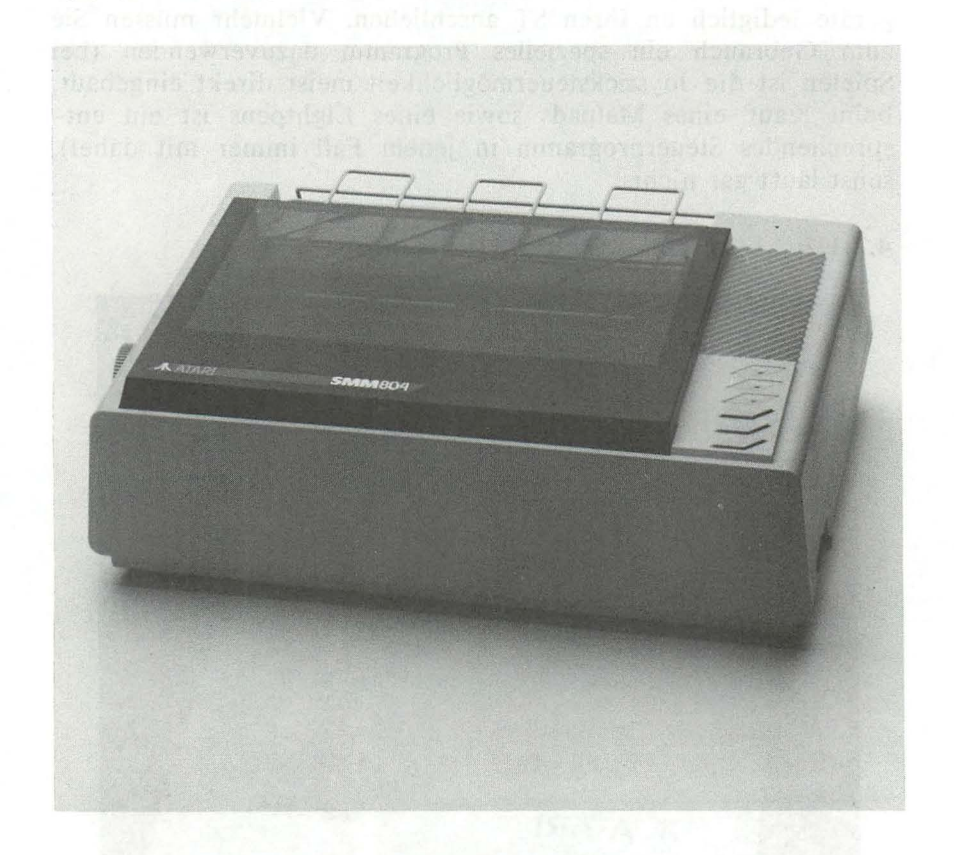
Aber: Es reicht in keinem Fall aus, wenn Sie eines der Steuergeräte lediglich an Ihren ST anschließen. Vielmehr müssen Sie zum Gebrauch ein spezielles Programm dazuverwenden (bei Spielen ist die Joysticksteuermöglichkeit meist direkt eingebaut, beim Kauf eines Malpads sowie eines Lightpens ist ein entsprechendes Steuerprogramm in jedem Fall immer mit dabei), sonst läuft gar nichts.

4.2.1.2 Anschluß für Diskettenstationen



Sie können an Ihren ST einen Speicher mit dem Diskettenformat 3.5 Zoll anschließen. Direkt mit der ersten Diskettenstation können Sie noch eine zweite Station verbinden. Wie das genau geht, können Sie in Kapitel 1.3 nachlesen. Die von ATARI für den ST angebotenen Diskettenstationen haben eine Speicherkapazität von 500 KByte bzw. 1 MByte.

4.2.1.3 Der Centronics-Anschluß



Für den Anschluß eines Druckers ist ein sogenannter Centronics-Parallel-Port in den ST eingebaut.

Centronics ist der Name der Entwicklungsfirma dieses Anschlusses (übrigens auch eine große Druckerfirma). Was heißt aber Parallelanschluß?

Erinnern Sie sich noch an unseren Ausflug zu Bits und Bytes? 8 Bits, 8 Ziffern einer Binärzahl ergeben 1 Byte. Über den Centronics-Anschluß sendet der ST nun immer 8 Bits, also 8 Stromsignale (entweder an oder aus) an den Drucker, und dies immer

gleichzeitig, praktisch nebeneinander. Weil die Bits also nebeneinander gleichzeitig herausgesendet werden, spricht man von paralleler Übertragung.

Wir haben vorhin gelernt, daß alle Zahlen unseres Dezimalsystems sich auch als Binärzzahlen darstellen lassen. Genauso hat man auch die Buchstaben mit einer Art "Kennziffer" versehen, die dann als Binärzahl über den Parallel-Anschluß an den Drucker gesendet wird.

Es gibt auch noch die Möglichkeit der seriellen (nacheinanderfolgenden) Übertragung der Bits. Dort wird dann Bit für Bit nacheinander gesendet. Sicher können Sie sich vorstellen, daß die serielle Übertragung im Vergleich zur parallelen sehr viel langsamer funktioniert.

Also kann man mit Fug und Recht behaupten: Ihr paralleler Centronicsanschluß ermöglicht eine sehr schnelle Datenübertragung zwischen Computer und Drucker. Nicht zuletzt haben deshalb auch alle größeren Personal-Computer Centronicsanschlüsse.

Kurz noch etwas dazu, welche Druckertypen Sie denn hier anschließen können:

Bestimmt haben Sie schon einmal etwas über die Nadel- oder *Matrixdrucker* gehört, die heutzutage meistverkauften. Hier wird jeder Buchstabe, jedes Zeichen, jede Zahl von meist 8 Nadeln achtmal hintereinander in der jeweils übermittelten Folge zu Papier gebracht.

Sicherlich ist Ihnen schon klar, daß hinter diesem Nadelandrucksystem wieder die An-Aus-Zustandverschlüsselung verborgen liegt. Bei diesem System werden zwar nicht immer sehr saubere Buchstabenabbilder auf dem Papier erzeugt wie bei einer Schreibmaschine, dafür aber können Sie hier auch mit einer Druckgeschwindigkeit von meist 50 Zeichen pro Sekunde, oftmals viel mehr, rechnen. Dies hängt mit der geringen mechanischen Beanspruchung des Zurückholens bzw. Andrückens der Nadeln auf dem Papier zusammen. Mit einer gewöhnlichen

Schreibmaschine oder auch einem *Typenraddrucker* (der ähnlich wie eine Schreibmaschine arbeitet) wäre diese Geschwindigkeit nicht zu erreichen (es sei denn, Sie bezahlen dafür etliche 1000 DM), da sich die Typen bei dieser Geschwindigkeit schon in der ersten Sekunde verfangen würden - so schnell könnten Sie überhaupt nicht in ihre Ausgangsstellung zurückgelangen.

Neben dem Nadel- und dem Typenraddrucker gibt es noch Tintenstrahldrucker, Thermodrucker und Laserdrucker.

Der *Tintenstrahldrucker* arbeitet im Prinzip ähnlich wie der Nadeldrucker, nur werden hier anstelle der Nadeln winzige Düsen verwendet, durch die mit Hilfe eines aufwendigen Mechanismus feinste Tintentröpfchen punktweise (!) auf das eingespannte Papier gespritzt werden.

Vorteil dieser recht teuren Drucker: Sie können kaum etwas vom Andruck hören (während ein Matrix- oder gar ein Typenraddrucker manchmal wie ein Sägewerk klingt). Nachteil der Tintenstrahldrucker: Sie können keine Durchschläge erzeugen, denn so hart treffen die Tintentröpfchen nicht auf dem Papier auf.

Die *Thermodrucker* arbeiten noch eine Idee leiser als die Tintenstrahldrucker. Der Schreibkopf erzeugt in der Form des jeweils auszugebenden Buchstabens oder Zeichens Wärme, die auf ein entsprechend empfindliches Spezialpapier übertragen wird. Dieses wärmeempfindliche Spezialpapier hat nun die Eigenschaft, daß überall dort, wo Wärme auf das Papier auftrifft, eine Farbe angezeigt wird (je nach Art des Papiers kann das Ergebnis blau, schwarz oder auch rot aussehen).

Auch eine interessante und recht preisgünstige Technik. Jedoch bedenken Sie, daß das Spezialpapier sehr teuer ist - wenn Sie häufig drucken, wird dann Ihr eigentlich so günstiger Thermodrucker von Tag zu Tag teurer.

Schließlich noch der *Laserdrucker*. Hier werden ganze Seiten meist innerhalb von wenigen Sekunden mit Hilfe der modernen Lasertechnik erzeugt. Atari bietet schon jetzt einen Laserdrucker mit sehr guter Auflösung zu einem günstigen Preis an.

Wir haben festgestellt, daß wir an unseren ST durch den Standard-Centronics-Parallelausgang mehr oder weniger beliebige Drucker anschließen können. Doch damit ist es noch nicht getan.

Zwar ist die in Ihren ST bereits eingebaute Centronics-Schnittstelle standardisiert, so daß Sie dort nicht nur die ATARI-eigenen Drucker anschließen können, sondern auch Drucker anderer namhafter Firmen wie EPSON, PANASONIC, NEC oder STAR, aber dies ist leider nicht so mit dem Zeichensatz!

Was ist das, ein 'Zeichensatz'? Ihr ST arbeitet, wie bereits des öfteren erwähnt, rein mathematisch. Sogar die auf dem Drucker erscheinenden Zeichen werden in Zahlencodes verschlüsselt. So trägt der Buchstabe 'A' die Zahl 65, das Leerzeichen die Zahl 32 usw.

Um nicht ein allzu großes Wirrwarr unter den verschiedenen Druckerherstellern herbeizuführen, hat man sich bereits vor vielen Jahren auf einen Standardzeichensatz mit dem Namen ASCII geeinigt. Dieser Standard bezieht sich allerdings lediglich auf die Zeichendarstellung mit dem Zahlschlüssel 0 bis 127.

Wenn Sie sich also einen Drucker zulegen möchten, sollten Sie sich von Ihrem Fachhändler beraten lassen, welchen Sie benutzen können. Im Laufe der Zeit werden mit Sicherheit immer mehr Hersteller auch ST-kompatible (anschließbare) Drucker anbieten (wie z.B. jetzt schon der TAXAN KP 810 ST).

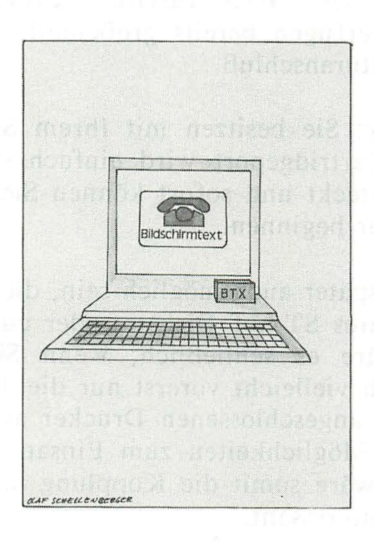
4.2.1.4 Der Cartridge-Anschluß

Schließlich verfügen alle ATARI ST über einen Erweiterungsanschluß, auch als Expansionanschluß oder Cartridgeport bezeichnet. Dieser Anschluß befindet sich an der linken Seite Ihres ST.

Ein Cartridge ist ein flaches Plastikkästchen, in dem eine kleine Platine mit einem ROM (na, wissen Sie noch, was das ist?) steckt. Meist werden Programme auf Cartridges den Disketten vorgezogen, denn: Sie stecken ein Cartridge einfach in den Cartridgeanschluß Ihres Computers ein, Sekunden später ist das Programm zum Arbeiten bereit (im Gegensatz zu einem Diskettenprogramm, bei dem Sie zumindest eine Vielzahl von Sekunden, manchmal auch einige Minuten warten müssen, bis Sie mit dem Arbeiten beginnen können).

Neben der Funktion des Cartridgeslots zum schnellen Einladen von Programmen in Ihren ST gibt es noch viel mehr Möglichkeiten.

Das BTX-Modul



BTX ist die Abkürzung für Bildschirmtext und repräsentiert das neueste Angebot der Deutschen Bundespost zur Steigerung der Kommunikationsmöglichkeiten (?!).

Um an diesem neuen Angebot der Post teilnehmen zu können, brauchen Sie einen Fernseher mit eingebautem BTX-Dekoder, einen Telefonanschluß und ein Post-BTX-Modem. Mit Hilfe der Fernbedienung Ihres Fernsehers ist es somit möglich, z.B. die aktuellen Informationsseiten Ihrer Tageszeitung auf dem Bildschirm erscheinen zu lassen oder Prospekte von Computerfirmen zu bestellen oder das eigene Konto via Bildschirm zu führen oder ... Die Möglichkeiten sind fast unbegrenzt groß.

Die Auswahl der BTX-Seiten funktioniert wie gesagt mit Ihrer Fernbedienung in Form von Zahlen. Wollen Sie allerdings an einen Hersteller eine bestimmte Frage über Bildschirmtext stellen oder einem Ihrer Verwandten (der natürlich auch an das BTX-Netz angeschlossen sein muß) einen Brief über BTX schreiben, müssen Sie, der Sie nur mit einer reinen Zahlenfernbedienung ausgestattet sind, passen.

Wollen Sie dies jedoch unbedingt tun, müssen Sie sich für einige 100 DM eine spezielle BTX-Tastatur dazukaufen, denn die BTX- Fernseher verfügen bereits großenteils über einen dafür vorgesehenen Tastaturanschluß.

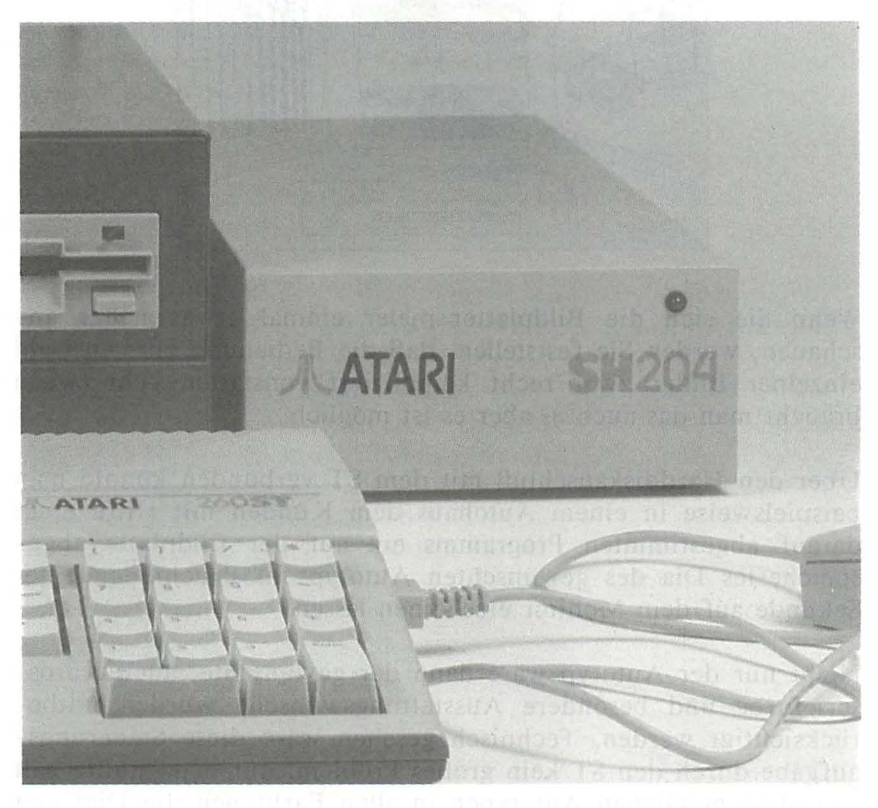
Nicht so bei Ihnen: Sie besitzen mit Ihrem ST ja bereits eine Tastatur. In den Cartridgeport wird einfach das entsprechende BTX-Modul eingesteckt und sofort können Sie mit der Eingabe von der ST-Tastatur beginnen.

Vielleicht wird es später auch möglich sein, die gesehenen BTX-Seiten mit Hilfe Ihres ST auf Diskette oder auf Harddisk abzuspeichern. Wie wäre es schließlich, wenn Sie die gesehenen Bilder - wenn auch vielleicht vorerst nur die Texte - direkt auf einen an Ihren ST angeschlossenen Drucker ausgeben könnten!? Neben den vielen Möglichkeiten zum Einsatz des ST zu Hause oder im Geschäft wäre somit die Kopplung mit BTX sicherlich auch für Sie sehr interessant.

Noch ist dies alles aber Zukunftsmusik. Wenn Sie an dieser neuen Technologie interessiert sind, sollten Sie die Fachpresse verfolgen. Vielversprechende Entwicklungen zum Thema BTX und ST laufen zur Zeit bei der Firma 'phs' in Hannover und bei 'Data Service' in Münster.

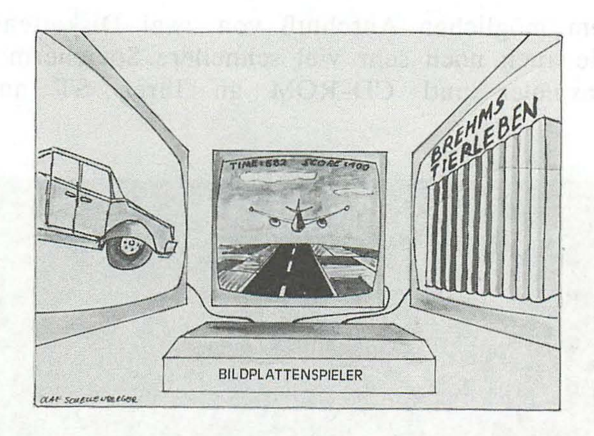
4.2.1.5 Der Harddisk-Anschluß

Neben dem möglichen Anschluß von zwei Diskettenstationen, werden Sie auch noch sehr viel schnellere Speichermedien wie Bildplattenspieler und CD-ROM an Ihren ST anschließen können.



Damit Sie einen Eindruck von der erzielten Schnelligkeit bekommen: Die Diskettenschnittstelle arbeitet mit 250000 Bit Datenübertragung pro Sekunde, die Harddiskschnittstelle aber mit 10000000 Bit Datenübertragung pro Sekunde (siehe BAUD).

Anschluß eines Bildplattenspielers



Wenn Sie sich die Bildplattenspieler einmal etwas näher anschauen, werden Sie feststellen, daß die Bedienung zur Auswahl einzelner Bilder zwar recht kompliziert vonstatten geht (wozu braucht man das auch?), aber es ist möglich.

Über den Harddiskanschluß mit dem ST verbunden könnte man beispielsweise in einem Autohaus dem Kunden mit Hilfe eines darauf abgestimmten Programms ein auf der Bildplatte abgespeichertes Dia des gewünschten Autotyps in Bruchteilen einer Sekunde auf dem Monitor erscheinen lassen.

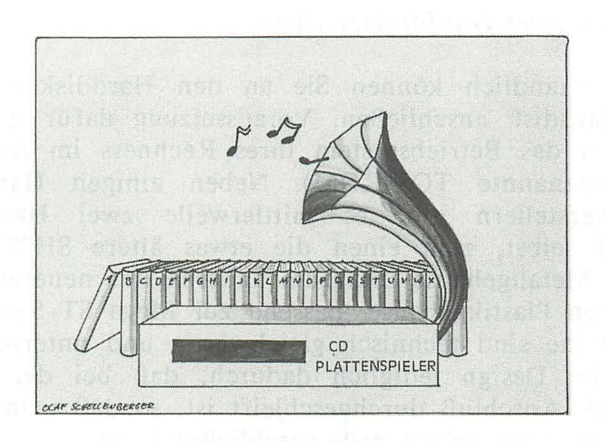
Nicht nur der Autotyp wäre dann der gewünschte, auch Karosseriefarbe und besondere Ausstattungswünsche würden mitberücksichtigt werden. Technisch gesehen wäre diese Steuerungsaufgabe durch den ST kein großes Problem, nur man müßte halt von allen möglichen Autotypen in allen Farbtönen die Dias erst einmal auf einer Bildplatte abspeichern, ein recht kostspieliges Verfahren, das sich erst z.B. für sehr große Autofirmen bezahlt machen würde.

Eine volkstümlichere Anwendung eines an Ihren ST angeschlossenen Bildplattenspielers wäre vielleicht Brehm's Tierleben: Sie bestimmen, welche Art von Tier Sie suchen (Säugetier, lebt

in Nordamerika ...); nachdem Ihr ST die Daten verarbeitet hat, steuert er den Bildplattenspieler an und wirft das gewünschte Dia in leuchtenden Farben auf den Bildschirm.

Schließlich eine dritte Anwendungsmöglichkeit für Bildplattenspieler am ST: Sie erfreuen sich an einem sogenannten Abenteuerspiel (ein Spiel, das aus einer buchähnlichen Handlung besteht). Immer dann, wenn eine interessante Aktion eintritt, wird Ihnen das entsprechende Bild von der Bildplatte angezeigt. Nicht nur das. Es könnte auch eine Sequenz von Bildern, ein Filmstück, angezeigt werden, in dem Sie, vielleicht mit Joystick in der Hand, die weitere Bildfolge selbst bestimmen können. Ein realistisches Telespiel, wie es kaum besser sein könnte.

Anschluß eines CD-Plattenspielers



Ähnlich könnten Sie mit einem CD-Plattenspieler, angeschlossen an Ihren ST, arbeiten: Nicht, daß Sie die gewünschten Tonsequenzen ansteuern wollten (vielleicht für Musiker ein interessanter Einsatz), es wäre doch denkbar, daß man solch eine CD-Platte ähnlich wie eine Diskette als Datenspeicher verwendet.

Während Sie auf einer Diskette 'nur' 180 DIN-A-4-Seiten abspeichern können, wären das auf einer CD-Platte 250000 DIN-A-4-Seiten.

Wie würde es Ihnen z.B. gefallen, wenn Sie demnächst über Ihren ST in Verbindung zum CD-Plattenspieler mal schnell im Brockhaus nachschlagen könnten?

Damit wir uns recht verstehen: Nicht im ST-Computer-RAM wären diese vielen Informationen abgespeichert, sondern auf der CD-Platte (bzw. beim Bildplattenspieler auf einer Bildplatte). Ihr ST steuert 'nur' das Auffinden der Daten; er liest alles in Sekundenschnelle durch bzw. verfügt über entsprechende Dateiordnungssysteme und steuert somit schließlich auch die Ausgabe auf dem Bildschirm erst dann an, wenn auch wahrlich die gewünschten Daten verfügbar sind.

Anschluß eines Harddiskspeichers

Selbstverständlich können Sie an den Harddiskanschluß auch eine Harddisk anschließen. Voraussetzung dafür ist allerdings, daß sich das Betriebssystem Ihres Rechners im Rom befindet (das sogenannte TOS-ROM). Neben einigen Harddisks von Fremdherstellern gibt es mittlerweile zwei Harddisks von ATARI selbst, zum einen die etwas ältere SH204 in einem grauen Metallgehäuse und zum anderen der neueren SH205 im modernen Plastikgehäuse, passend zur Mega ST-Serie. Die beiden Geräte sind technisch gleichwertig und unterscheiden sich außer im Design lediglich dadurch, daß bei der SH205 der Harddisk-Anschluß durchgeschleift ist, so daß man noch mehr Geräte an diese Schnittstelle anschließen kann.

Packen Sie die Harddisk vorsichtig aus, und nehmen Sie alle Teile aus dem Karton. Sie sollten nun neben einer Bedienungsanleitung die Festplatte selbst, ein Netzkabel, ein Kabel mit zwei breiten Steckern und eine Diskette vor sich haben. Bevor wir jetzt mit der Installation der Festplatte beginnen, fertigen Sie bitte zunächst eine Sicherheitskopie der Diskette an. Stellen Sie die Festplatte zunächst so hin, wie sie später stehen soll. Die SH205 kann hervorragend als Monitorständer benutzt werden, beim Mega ST dient sie als Unterlage für den Rechner. Bei ausgeschaltetem Rechner verbinden Sie nun das Kabel mit den

breiten Steckern auf der einen Seite mit Buchse 3 (Kapitel 1.1), die die Aufschrift Harddisk trägt, auf der anderen Seite mit der Buchse mit der Bezeichnung 'Harddisk ein' bei der SH205 bzw. mit der einzigen Buchse bei der SH204. Schrauben Sie das Kabel mit jeweils zwei Schrauben fest, damit es sich nicht mehr lösen kann. Nun können Sie das Netzkabel in die entsprechende Buchse der Harddisk stecken. Überprüfen Sie noch einmal alle Verbindungen. Ist alles in Ordnung, kann es losgehen.

Bevor Sie mit der Harddisk arbeiten können, muß diese noch darauf vorbereitet werden. Sie sollten auf jeden Fall das Kapitel 'Arbeiten mit dem ST' bereits durchgearbeitet haben, damit Sie jetzt keine Probleme mehr mit dem Starten von Programmen haben.

Legen Sie Sicherheitskopie der Boot-Diskette in Ihr Laufwerk A. Schalten Sie nun die Festplatte ein (Der Schalter befindet sich hinten an der Festplatte). Nun sollte der Ventilator zu hören sein, ein grünes Lämpchen ganz und ein rotes Lämpchen leicht flackernd angehen. Wenn nicht, überprüfen Sie, ob das Netzkabel richtig und auch in der Steckdose steckt. Nach einigen Sekunden wird das rote Lämpchen ausbleiben. Schalten Sie jetzt Ihren Computer ein. Nach kurzer Zeit sollte sich das Desktop mit einem weiteren Diskettensymbol C melden. Klicken Sie die Station A an, und lassen Sie sich den Inhalt der Diskette anzeigen. Dort werden Sie einen Auto-Ordner sehen, in dem sich ein Treiberprogramm für den Festplattenbetrieb befindet. Ohne diesen Treiber ist ein Arbeiten mit Festplatte nicht möglich.

Auf der Diskette sollte auch ein Programm mit dem Namen 'HDX.PRG' sein. Starten Sie dies durch Doppelklick. Das Programm ist eine kleine Harddisk-Utility, mit der Sie die Platte für die Arbeit vorbereiten können. Dieser Schritt ist nur einmal am Anfang notwendig, Sie können aber auch wie bei Disketten mit der folgenden Prozedur die gesamte Festplatte löschen.

Sie haben nun eine Menüzeile mit drei Menüpunkten. Unter dem Menüpunkt 'Desk' versteckt sich eine kleine Info-Box, mit dem Menüpunkt 'File' können Sie durch 'Quit' das Programm beenden. Interessanter ist der Menüpunkt 'Disk'. Ähnlich wie bei einer Diskette muß die Festplatte zuerst einmal formatiert werden. Klicken Sie dazu den Punkt 'Format' an. Eine Warnbox gibt Ihnen die Gelegenheit abzubrechen, mit 'OK' kommen Sie in

eine Box, in der Sie eine Unit wählen können. Da Sie vermutlich nur eine Festplatte haben, klicken Sie den Punkt 'UNIT 0' an und bestätigen mit 'OK'. In der nächsten Box werden Sie nach dem Namen Ihrer Festplatte gefragt. Klicken Sie hier Ihren Typ an und bestätigen mit 'OK'. So nun haben Sie es gleich geschafft. Nach einer letzte Sicherheitsabfrage, die mit 'YES' zu beantworten ist, wird die Festplatte endlich formatiert.

Der nächste Schritt ist das sogenannte Partitionieren der Festplatte. Der ATARI ST kann nur maximal 16 MB große Laufwerke ansprechen. Da wir jedoch 20 MB zur Verfügung haben und auch nutzen wollen, müssen wir die Festplatte in mehrere Teile, sogenannte Partitionen, aufteilen. Diese Partitionen können wir später wie verschiedene Floppy-Laufwerke ansprechen. Klicken wir also 'Partition' an und übergehen die Sicherheitsabfrage mit 'OK'. Auch hier müssen wir wieder die Unit angeben. Wählen Sie Unit 0 und machen weiter mit 'OK'. Nun haben Sie die Möglichkeit, maximal vier Partitionen einzustellen. Beim Punkt Menü sind schon Vorschläge eingetragen. Entscheiden Sie sich für eine fertige Größe, oder editieren Sie die Partitionsgrößen nach Wunsch. Sie können auch einzelne Partitionen ganz ausschalten, indem die Größe auf 0 gestellt wird. Bitte beachten Sie, daß die Summe der Partitionen maximal 20 MB erreicht. Das ist bei der Voreinstellung nicht der Fall, verkleinern Sie daher die 1. Partition auf 0 und vergrößern Sie sie dann auf 4 MB. Wenn Sie damit fertig sind, klicken Sie das 'OK'-Feld an. Nun werden die einzelnen Partitionen auf der Festplatte angelegt.

Sie haben sicher gesehen, daß noch zwei weitere Punkte auf der Liste stehen. Obwohl diese nicht unbedingt nötig sind, sollten Sie sich doch die Arbeit machen, beides durchzuführen. Als erstes betätigen wir den Punkt 'Zero'. Wie der Name schon sagt (Zero bedeutet Null), werden damit einzelne Partitionen mit Nullen vollgeschrieben. Damit ist die Partition vollständig gelöscht. Auch hier übergehen wir die Sicherheitsabfrage mit 'OK'. Klicken Sie nun zuerst die erste Partion C und das 'OK'-Feld an. Die Partition wird jetzt mit Nullen vollgeschrieben. Wiederholen Sie diesen Punkt für alle angemeldeten Partionen.

Jetzt kommen wir zum letzten Punkt. Mit der Funktion 'Markbad' werden die schlechten Sektoren auf einer Partition gekennzeichnet. Wie bei schlechten Disketten kommt es auch bei Festplatten vor, daß ein Sektor nicht richtig beschreibbar ist.

Damit diese später nicht zufällig benutzt werden, werden diese Sektoren gekennzeichnet und stehen nicht für die Speicherung zur Verfügung. Ein defekter Sektor pro MB ist ein normaler Wert, d.h. auf unserer gesamten Festplatte könnten etwa 20 Sektoren defekt sein. Die Funktion 'Markbad' wird genau wie die Funktion 'Zero' mit allen angemeldeten Partitionen durchgeführt.

So, nun haben Sie es endlich geschafft. Verlassen Sie das Programm mit 'Quit', und klicken Sie die Hard-Disk C an. Augenblicklich erscheint ein leeres Inhaltsverzeichnis, denn auf der Platte ist ja noch nichts gespeichert. Schließen Sie zunächst alle Fenster, und schauen Sie sich Ihr Desktop an. Obwohl wir weitere Partitionen angegeben haben, sind diese nirgendwo zu sehen. Um dem abzuhelfen, klicken Sie das Symbol C an, so daß es revers erscheint, und klicken im Menü 'Extras' den Punkt 'Floppy anmelden...' an. In der Box verändern wir den Buchstaben 'C' in ein großes 'D' und klicken 'Anmelden' an. Auf dem Bildschirm ist jetzt das Symbol für die Partion D aufgetaucht. Wiederholen Sie diesen Vorgang, bis alle Ihre angemeldeten Partitionen auf dem Bildschirm aufgetaucht sind. Damit Sie diese Anmelderei nicht jedesmal neu durchführen müssen, speichern Sie sie mit 'Arbeit sichern' auf die Sicherheitskopie ab. Wenn Sie beim nächsten Mal mit dieser Diskette booten, werden die Festplatten-Symbole automatisch vorhanden sein.

Mit der Festplatte können Sie jetzt so arbeiten, wie Sie es von Ihren Disketten gewohnt sind, jedoch um einiges schneller. Versuchen Sie es, indem Sie einfach die Programme Basic.prg und Basic.rsc auf die Festplatte kopieren und dort starten.

Damit Ihre Festplatte arbeitet, müssen Sie unbedingt die Sicherheitskopie Ihrer Boot-Diskette ins Disklaufwerk A einlegen, bevor Sie den Rechner einschalten. Auf den neueren Boot-Disketten gibt es jetzt ein Programm, das es ermöglicht, von der Festplatte zu booten. Dieses Programm trägt den Namen 'Hinstall.prg'. Nachdem Sie es gestartet haben, können Sie im Menü 'File' mit 'Install' nach der Sicherheitsabfrage die Partition C bootbar machen. Kopieren Sie jetzt die Datei 'Desktop.inf' vom Laufwerk A auf die Partition C. Denken Sie daran, daß auch Accessories nun nur noch von der Festplatte gestartet werden. Kopieren Sie die notwendigen also ebenfalls auf die Partition C.

Wichtiger Hinweis: Wie bereits beschrieben, muß beim Einschalten der Geräte immer zuerst die Festplatte eingeschaltet werden. Die Festplatte initialisiert sich selbst und braucht dazu etwa 10 bis 15 Sekunden. In dieser Zeit darf sie nicht durch ein Signal vom Rechner gestört werden, sonst funktioniert die Festplatte nicht. Sie erkennen es daran, daß die Symbole für die Festplatte fehlen oder die rote Lampe der Festplatte nicht mehr ausgeht. Also: immer zuerst die Harddisk einschalten, 15 Sekunden warten, erst dann Rechner und Monitor einschalten.

4.2.1.6 Die MIDI-Anschlüsse

Die MIDI-Anschlüsse sind dafür gedacht, daß Sie eine oder mehrere Synthesizer oder fernsteuerbare Orgeln an Ihren ST anschließen können. Sie können diese Verbindung über die genormten MIDI-Anschlüse vornehmen. Dabei kommt Ihrem ST die Funktion zu, sich verschiedene Vorgänge, die am Synthesizer vorher eingegeben wurden, zu merken (z.B. welche Stimme auf dem Synthesizer eingestellt wurde, oder welche Melodie gespielt wurde) und diese Tätigkeiten nachher automatisch wieder und wieder abzuwickeln. So könnten Sie nach der Einspeicherung über ein Menü auf dem Bildschirm Ihres ST mit einem Tastendruck auswählen, welche Sequenz Ihre Orgel spielen soll, und vielleicht auch noch mit welcher Stimme.

4.2.1.7 Die RS-232-Schnittstelle

Im Gegensatz zur MIDI-Schnittstelle, die zumindest von Ihrer eigentlichen Bedeutung her nur für einen kleinen Teil der ST-Besitzer geeignet ist, verhält sich das mit der RS-232-Schnittstelle ganz anders.

Sie können über diese Schnittstelle z.B. die Verbindung zu einem anderen Computer aufnehmen (der auch über eine RS-232-Schnittstelle verfügen muß) und mit Hilfe eines geeigneten Programms Daten zwischen beiden Geräten austauschen.

Außerdem ist es möglich, über die RS-232-Schnittstelle auch Verbindung mit einem Akkustikkoppler über das Telefonnetz nach draußen hin vorzunehmen. Ein Akustikkoppler ist ein Gerät, das die elektrischen Ströme in Ihrem Computer in Pieps-

töne umwandeln kann (codieren), und diese an der anderen Seite auch wieder in elektrische Ströme zurückverwandeln kann (decodieren). Man nennt diesen Vorgang auch modulieren und demodulieren - daher die andere Bezeichnung für einen Akustikkoppler: Modem. Diese Technik muß angewendet werden, um Daten per Telefonleitung von einem Computer zum anderen zu übertragen, denn unsere Telefonleitungen können nicht direkt elektrische Ströme von einem Computer zum anderen weiterleiten, wohl aber verschlüsselte Ströme in Form von den besagten Piepstönen.

Schließlich ist es möglich, daß Sie über die RS-232-Schnittstelle einen *Drucker* an Ihren ST anschließen. Allerdings können hier nur Drucker angeschlossen werden, die ebenfalls über solch eine Schnittstelle verfügen.

Interessant für Firmen in diesem Zusammenhang ist sicher die Tatsache, daß der ATARI ST auch als Terminal arbeiten kann. Hierbei arbeitet der ST nur als Empfänger und Sender von Daten. Der Zentralrechner kann in weiterer Entfernung stehen und von vielen Benutzern angesprochen werden. Der normale Betrieb des ST als eigenständiger Rechner kann dann auf Knopfdruck eingeschaltet werden, sodaß zwischen Terminalbetrieb und ST-Modus umgeschaltet werden kann.

Was unterscheidet die RS-232-Schnittstelle von der Centronics-Druckerschnittstelle? Die Centronics-Schnittstelle übermittelt die Daten parallel, d.h. jeweils 8 Bit oder 1 Byte mit einem Mal. Bei der RS-232-Schnittstelle sprechen wir von einer seriellen Schnittstelle, denn sie übermittelt die Daten (Bit für Bit) in Serie, also ein Bit nach dem anderen. Deshalb kann man auch sagen, daß die RS-232-Schnittstelle in Ihrer Datenübertragungsleistung langsamer arbeitet als die Centronics- Schnittstelle.

4.2.1.8 Der Monitoranschluß

Die Monitorbuchse Ihres ST ist schlau. Sie erkennt automatisch, ob Sie einen Schwarz/Weiß-Monitor oder einen Farbmonitor angeschlossen haben.

Ist es ein Farbmonitor, so werden Ihnen die grafischen Auflösungsstufen 640*200 mit 4 Farben und 320*200 mit 16 Farben

angeboten, ist es aber ein Schwarz/Weiß-Monitor (der bekanntlich höher auflöst als ein Farbmonitor), so steht Ihnen 'nur' die höchste Auflösungsstufe mit 640*400 Punkten in 2 Farben zur Verfügung.

Zudem ist es möglich, mittels eines besonderen Kabels die Verbindung vom ST zu einem Fernseher mit Scart-Anschluß vorzunehmen. Klar, daß diese Verbindung zur Folge hat, daß Ihr ST in den Farbmodus springt und die zur Verfügung stehende Farbpalette und die dazu passende Grafikauflösung automatisch anwählt. Das gleiche gilt beim ATARI 520 ST/M im Fernsehbetrieb.

4.2.1.9 Der Netzanschluß

Schließlich ist da noch ein Stecker für die Stromzuführung, wobei Sie nur darauf achtgeben sollten, daß dieser Stecker immer fest in Ihren Computer eingeschoben ist - sonst ist bei unerwartetem Computerausfall und dem daraus resultierendem Löschen der Daten und Programme das Wehklagen sehr groß!

Vielleicht haben Sie in diesem Abschnitt nicht nur erfahren, daß die vielen unscheinbaren Buchsen rund um Ihren ST besondere Funktionen erfüllen, sondern auch ein wenig mehr darüber Kenntnis erworben, wie Sie Ihren ST zukünftig sinnvoll einsetzen können.

4.2.2 Weiche Aussichten

Nach diesem Ausflug in die Hardware-Welt (Hardware = alles was Sie anfassen können, also die Geräte usw.) wollen wir uns nun der Software (= weiche Ware, also die Programme und Daten) zuwenden, die für Ihren ATARI ST zu haben ist bzw. sein wird. Mit verschiedenen Programmiersprachen haben wir uns ja schon beschäftigt. Es gibt jedoch auch fertige Programme, z.B. Anwenderprogramme.

Für einen derart erfolgreichen Computer, wie es der ATARI ST von Anfang an war, bieten fast alle Softwarefirmen Programme an. So gibt es Textverarbeitungs-, Dateiverwaltungs-, Finanz-

buchhaltungs-, Zeichen-, Mal- und Spieleprogramme in rauhen Mengen. Hier müssen Sie wohl oder übel die Wahl selbst treffen. Hilfreich ist dabei sicher, die Fachpresse zu verfolgen.

Es gibt dabei Programme, die mit der Maus und den grafischen Möglichkeiten des GEM arbeiten, und solche, die diese Möglichkeiten außer acht lasssen. Sie sollten sich daher im Laden vorführen lassen, was Sie da kaufen, damit Sie sich schon vorher ein Bild machen können.

Von DIGITAL RESEARCH, dem Entwickler von GEM, gibt es z.B. eine Textverarbeitung (GEM WRITE) und Malprogramme (GEM DRAW), die die Fähigkeiten des GEM voll auskosten. Aber auch andere Hersteller bieten exzellente Programme an. Viele Spieleprogramme und auch Lernsoftware gibt es bereits, und von Tag zu Tag kommen ein paar mehr auf den Markt.

Ob nun Hardware oder Software, Sie sind durch den Besitz Ihres ATARI ST in der Lage, ein Rechnersystem zu benutzen, dem die Zukunft gehören wird. Um Ihnen dies noch einmal anschaulich zu machen, wollen wir uns zusammen noch kurz zusammenfassend ein paar Gedanken über den ATARI ST machen.

4.3 Bedeutung des ATARI ST

Als Jack Tramiel, Chef von ATARI, im Januar 1985 den ATARI-Stand der CES (Computermesse) in Las Vegas mit großem Brimborium eröffnete, sagte er: "Wir werden keine Home-Computer verkaufen. Wir werden keine Business-Computer verkaufen. Aber wir werden Personal-Computer verkaufen."

Doch was heißt denn das schon: "Personal-Computer"? Vor kurzem hielt ich einen kleinen Taschencomputer einer japanischen Firma in der Hand, der ebenfalls die Bezeichnung Personal-Computer trägt. So war es bisher gewesen: Da es nie eine richtige Vereinbarung darüber gegeben hat, was Personal-Computer für Kriterien erfüllen müssen, war dieser Begriff nicht mit Inhalt gefüllt worden. Sollte ein Personal-Computer ein Gerät für das 'Personal' einer Firma sein oder der 'persönliche' Computer, den man bei sich zu Hause verwahrt oder gar einer, den man in die Jackentasche stecken kann?

Jack Tramiel hat seine Personal-Computer mit dem Schlagwort 'Power without the price', oder übersetzt 'Wir machen Spitzentechnologie preiswert' vorgestellt. So viel man auch über den Begriff 'Personal-Computer' streiten kann, so eindeutig ist diese ATARI-Maxime.

Was ist neu an der ST-Serie von ATARI? Erst einmal haben sie von vornherein so viel Speicherplatz, wie es ihn bisher nur auf Diskettenstationen gab - ab 512 Kilobyte aufwärts. Außerdem glänzen sie durch den Einbau des 16-Bit-Prozessors Motorola-68000 mit einer unvorstellbaren Verarbeitungsgeschwindigkeit. Was haben Sie davon als Anwender? Statt 100000 Rechenoperationen pro Sekunde kann Ihr ST mit Hilfe dieses Prozessors 1000000 Berechnungen pro Sekunde ausführen!

Wofür braucht man denn solch eine Geschwindigkeit? Denken Sie einmal daran, wie groß ein Text sein kann, der in Ihren ST paßt. Nun suchen Sie eine ganz bestimmte Stelle und geben Ihrem Computer den Befehl, diese Stelle zu finden. Da jeder Computer jede Aufgabe, die Sie an ihn stellen, auf mathematischem Wege löst, können Sie sich nun vielleicht an diesem Beispiel vorstellen, wofür schnellere Computer sinnvoll sind! Denken wir gar noch einen Schritt weiter: Ihr ST soll in einem auf CD-Platte abgespeicherten Lexikon nachschauen, in welchem Artikel etwas über Computer steht...

Doch das ist noch nicht alles. Die ST-Palette umfaßt viele Geräte, die Sie an Ihren ST anschließen können und die über ein bisher nie dagewesenes Preis-Leistungs-Verhältnis verfügen wie z.B. eine Diskettenstation mit 1 Megabyte Speicherplatz für weniger als 500 DM oder eine Harddisk mit 20 Megabyte für unter 1300 DM ... halt 'Power without the price'.

Außerdem verfügt Ihr ATARI ST über eine Menge Standardanschlüsse wie z.B. zur Verbindung mit anderen Computern, fast beliebigen Druckern und sogar mit professionellen Musikinstrumenten. Schließlich seine grafischen Fähigkeiten, mit denen er jeden Konkurrenten aussticht: 512 Farben und eine hoch auflösende Grafik mit bis zu 640*400 Bildpunkten sind darstellbar.

Noch ein letzter Punkt, warum der ST so außergewöhnlich ist: Bisher war es fast unmöglich, daß sich ein Computerneuling mit

solch einem Wunderkästchen ohne Probleme beschäftigen konnte. Dies lag bzw. liegt großenteils darin begründet, daß das Arbeiten mit sinnvollen Programmen erst einmal einen Rattenschwanz von zusätzlichen und für den Laien anscheinend sinnlosen Tätigkeiten erfordert ('mit welchen Befehlen muß ich ein Programm starten? ...'). Jack Tramiel hat auch hier die richtige Nase gehabt und gibt uns, dem Computerbenutzer, gleich ein Hilfsmittel mit an die Hand, so daß wir uns an Bildsymbolen ganz einfach orientieren können. Das nennt man ein wahres Denken im Sinne des eigentlichen Computerbenutzers!

Vielleicht denken Sie nun: da müssen ja Jahrzehnte vergangen sein, bis diese Spitzentechnologie fertiggestellt wurde. Sicherlich, einerseits haben Sie recht mit dieser Annahme, denn in Ihrem ST stecken natürlich etliche Errungenschaften der Technik verborgen, die eine lange Entwicklungszeit hinter sich haben. Jack Tramiel und ATARI erkannten aber als erste Mitte 1984, daß nun die Zeit gekommen war, diese Spitzentechnologie preiswert anbieten zu können. Mitte 1984 begann nämlich die Entwicklungsarbeit für den ST ... übrigens auf einem gewöhnlichen Küchentisch. Anfang 1985 wurde der ST dann auf der CES in Las Vegas dem staunenden Publikum vorgestellt, und seit Mitte 1985 wird er nun in riesigen Stückzahlen produziert und findet nicht nur in Deutschland reißenden Absatz.

Welchen Stellenwert nimmt Deutschland als Computermarkt in den Augen von Jack Tramiel und ATARI ein? Schauen wir uns an, wo ATARI seit der CES in Las Vegas bis Ende 1985 den ST auf Messen vorgeführt hat: April 85: Hannover Messe; Juni 85: CES, Chicago; September 85: PCW, London; Oktober 85: Systems, München. Braucht man einen besseren Beweis dafür, daß Jack Tramiel neben dem amerikanischen Computermarkt den in Deutschland besonders hoch einschätzt? Von fünf Messebesuchen waren immerhin zwei in Deutschland und 1986 ging es so weiter: Im März 1986 die Hannover Messe und ...

Eine neue Entwicklung wirft allerdings auch Probleme auf für diejenigen, die über das Produkt Bücher schreiben. So war es uns während der Schreibphase nicht immer bekannt, wie ATARI in Zukunft sein Sortiment (Software und Hardware) erweitern wird. Dies ist der Grund dafür, daß Sie ab und zu vielleicht feststellen werden, daß Aussagen, die wir heute zu den Geräten machen, in dem Moment, in dem Sie das Buch lesen, möglicher-

weise schon überholt sind. Gerade was die Ausstattung der Geräte betrifft, hat ATARI in der Vergangenheit bewiesen, daß Überraschungen vorgeplant sind, und dies wird für die Zukunft sicher in noch größerem Maße zutreffen. Wir stehen schließlich erst am Anfang der ST-Technologieentwicklung. Viel Neues wird sich noch tun, was man heute nur ahnen und darum nicht mit Sicherheit sagen kann.

Sollte dies ein Buch leisten, so dürfte es eigentlich nie erscheinen, denn 'lebende' Computerfirmen sind 'lebendig' (sonst würden sie nicht sehr lange leben) und lassen sich somit des öfteren etwas Neues einfallen.

Um daher immer "am Ball" zu bleiben, sollten Sie sich regelmäßig eine Computerzeitschrift kaufen, um zu verfolgen, was sich so tut. In der DATA WELT, die monatlich erscheint, finden Sie daneben auch noch jede Menge Tips & Tricks für Ihren ATARI ST, Programme, Tests, usw. Sie haben dort auch die Möglichkeit, bei Problemen oder Fragen direkt mit der Redaktion Kontakt aufzunehmen.

Apropos Kontakt. Natürlich können Sie auch mit uns, den Schreiberlingen dieses Buches, Kontakt aufnehmen, wenn Sie Fragen, Probleme, Kritik oder Verbesserungsvorschläge zu diesem Buch haben:

Rainer Lüers
Tel. 0251/47478
(BTX-Anschluß und Anrufbeantworter)

Michael Stein
Postfach 10 24 45
4630 Bochum 1

Im Anhang finden Sie nun noch einiges Wissenswerte zu den Tastaturbelegungen, Umrechnungen in die verschiedenen Zahlensysteme und ein Mini-Lexikon der Computerei, wo Fachbegriffe kurz erklärt werden. Ganz am Schluß finden Sie noch das Stichwortverzeichnis.

5. Anhang

5.1 ATARI ST-Zeichensatz

Wie Sie sehen, hat Ihr ST einen größeren Zeichensatz, als wir mit der Tastatur direkt ansteuern können.

```
Desk  File  Run  Edit  Debug
                         OUTPUT

        Der Zeichensatz des ATARI ST-Computers im ST-BASIC
        --------------------------------------------------
0    1 ⇧  2 ⇩  3 ⇨  4 ⇦  5 ☒  6 ▣  7    8 ✓  9 ◷
10
 11 ♪  12 ␌  13    14 ◢  15 ◣  16 0  17 1  18 2  19 3
20 4  21 5  22 6  23 7  24 8  25 9  26 ə  27 ␛  28 ┌  29 ┐
30 ³  31 ⁷  32    33 !  34 "  35 #  36 $  37 %  38 &  39 '
40 (  41 )  42 *  43 +  44 ,  45 -  46 .  47 /  48 0  49 1
50 2  51 3  52 4  53 5  54 6  55 7  56 8  57 9  58 :  59 ;
60 <  61 =  62 >  63 ?  64 @  65 A  66 B  67 C  68 D  69 E
70 F  71 G  72 H  73 I  74 J  75 K  76 L  77 M  78 N  79 O
80 P  81 Q  82 R  83 S  84 T  85 U  86 V  87 W  88 X  89 Y
90 Z  91 [  92 \  93 ]  94 ^  95 _  96 `  97 a  98 b  99 c
100 d  101 e  102 f  103 g  104 h  105 i  106 j  107 k  108 l  109 m
110 n  111 o  112 p  113 q  114 r  115 s  116 t  117 u  118 v  119 w
120 x  121 y  122 z  123 {  124 |  125 }  126 ~  127 ∆  128 Ç  129 ü
```

```
Desk  File  Run  Edit  Debug
                         OUTPUT

        Der Zeichensatz des ATARI ST-Computers im ST-BASIC
        --------------------------------------------------
130 é  131 â  132 ä  133 à  134 å  135 ç  136 ê  137 ë  138 è  139 ï
140 î  141 ì  142 Ä  143 Å  144 É  145 æ  146 Æ  147 ô  148 ö  149 ò
150 û  151 ù  152 ÿ  153 Ö  154 Ü  155 ¢  156 £  157 ¥  158 ß  159 ƒ
160 á  161 í  162 ó  163 ú  164 ñ  165 Ñ  166 ª  167 º  168 ¿  169 ⌐
170 ¬  171 ½  172 ¼  173 ¡  174 «  175 »  176 ã  177 õ  178 Ø  179 ø
180 œ  181 Œ  182 À  183 Ã  184 Õ  185 ¨  186 ´  187 †  188 ¶  189 ©
190 ®  191 ™  192 ĳ  193 Ĳ  194 א  195 ב  196 ג  197 ד  198 ה  199 ו
200 ז  201 ח  202 ט  203 י  204 כ  205 ל  206 מ  207 נ  208 ס  209 ע
210 פ  211 צ  212 ק  213 ר  214 ש  215 ת  216 ן  217 ך  218 ם  219 ף
220 ץ  221 §  222 ∧  223 ∞  224 α  225 β  226 Γ  227 π  228 Σ  229 σ
230 µ  231 τ  232 Φ  233 Θ  234 Ω  235 δ  236 ∮  237 ϕ  238 ∈  239 ∩
240 ≡  241 ±  242 ≥  243 ≤  244 ⌠  245 ⌡  246 ÷  247 ≈  248 °  249 •
250 ·  251 √  252 ⁿ  253 ²  254 ³  255 ¯
```

Wie lassen sich nun die vielen Sonderzeichen ansteuern? Einerseits haben Sie die Möglichkeit, sich die einzelnen Zeichen mit 'PRINT CHR$(...)' in BASIC auf dem Bildschirm ausgeben zu lassen. Auf der anderen Seite gibt es auch noch eine andere Möglichkeit, einen Großteil dieser Sonderzeichen direkt per Tastatur zu erzeugen:

1:	Control-Ü und Control-A	19:	Control-S und Control-3
2:	Control-B	20:	Control-T und Control-Ö und Control-4
4:	Control-D und Control-Ä		
5:	Control-E	21:	Control-U und Control-5
6:	Control-F	22:	Control-V
9:	TAB-Taste und Control-I	23:	Control-W und Control-7
11:	Control-+ und Control-K	24:	Control-X und Control-8
12:	Control-L und Control-,	25:	Control-Y und Control-9
14:	Control-N und Control-.	27:	Esc-Taste
15:	Control-O	28:	Control-<
16:	Control-P und Control-0	29:	Control-Shift-0
17:	Control-Q und Control-1	30:	Control-6 und Control-ß
18:	Control-R	31:	Control--

Wollen Sie z.B. das ATARI-Zeichen auf dem Bildschirm ausgeben, haben Sie zwei Möglichkeiten:

1. PRINT CHR$(14);CHR$(15)
2. Control-N und dann Control-O drücken

5.2 Umrechnungsprogramme

1. Umrechnung Dezimal - Binär

```
10 INPUT "Dezimalzahl";A
20 FOR N=15 TO 0 STEP-1
30 IF A=INT(2^N) THEN A$=A$+"1":A=A-INT(2^N):Z=1
40 IF A>INT(2^N) THEN A$=A$+"1":A=A-INT(2^N):Z=1
50 IF Z=0 THEN A$=A$+"0"
60 Z=0:NEXT N
70 PRINT A$
```

Aufgabe: Umrechnung von 255 Dezimal in die entsprechende Binärzahl

```
RUN
Dezimalzahl ? 255
Ergebnis: 0000000011111111
```

2. Umrechnung Binär - Dezimal

```
10 INPUT "Binaerzahl";A$
20 FOR N=1 TO LEN(A$)
30 IF MID$(A$,N,1)="1" THEN Z=1
40 IF Z=1 THEN Z=0:A=A+2^(LEN(A$)-N)
50 NEXT N
60 PRINT A
```

Aufgabe: Umrechnung von 11111111 Binär in die entsprechende Dezimalzahl

```
RUN
Binaerzahl? 11111111
Ergebnis: 255
```

3. Umrechnung Dezimal - Hexadezimal

Aufgabe: Umrechnung von 255 Dezimal in die entsprechende Hexadezimalzahl

```
PRINT HEX$(255)
Ergebnis: FF
```

4. Umrechnung Hexadezimal - Dezimal

Aufgabe: Umrechnung von FF Hexadezimal in die entsprechende Dezimalzahl

```
PRINT &HFF
Ergebnis: 255
```

5. Umrechnung Dezimal - Oktal

Aufgabe: Umrechnung von 255 Dezimal in die entsprechende Oktalzahl

```
PRINT OCT$(255)
Ergebnis: 377
```

6. Umrechnung Oktal - Dezimal

Aufgabe: Umrechnung von 377 Oktal in die entsprechende Dezimalzahl

```
PRINT &O377
Ergebnis: 255
```

7. Umrechnung Binär - Hexadezimal

```
10 INPUT "Binaerzahl";A$
20 FOR N=1 TO LEN(A$)
30 IF MID$(A$,N,1)="1" THEN Z=1
40 IF Z=1 THEN Z=0:A=A+2^(LEN(A$)-N)
50 NEXT N
60 A%=A
70 PRINT HEX$(A%)
```

Aufgabe: Umrechnung von 11111111 Binär in die entsprechende Hexadezimalzahl

```
RUN
Binaerzahl? 11111111
Ergebnis: FF
```

8. Umrechnung Hexadezimal - Binär

```
10 INPUT "Hexadezimalzahl";A$
20 A$="&H"+A$
30 A=VAL(A$):A$=""
40 FOR N=15 TO 0 STEP-1
50 IF A=INT(2^N) THEN A$=A$+"1":A=A-INT(2^N):Z=1
60 IF A>INT(2^N) THEN A$=A$+"1":A=A-INT(2^N):Z=1
70 IF Z=0 THEN A$=A$+"0"
80 Z=0:NEXT N
90 PRINT A$
```

Aufgabe: Umrechnung von FF Hexadezimal in die entsprechende Binärzahl

```
RUN
Hexadezimalzahl ? FF
Ergebnis: 0000000011111111
```

5.3 Mini-Lexikon der Computerei

Akustikkoppler: Gerät zur Übertragung von Daten per Telefon.

BASIC: Programmiersprache, die verhältnismäßig leicht zu erlernen ist (weil sofort Fehler erkannt werden und man sich die Befehle gut merken kann).

Baud: Geschwindigkeit, mit der Bits pro Sekunde vom Computer zur Peripherie übertragen werden.

Betriebssystem: Programm, das die Steuerung zwischen Computer und Peripherie regelt.

Binärsystem: Der Computer arbeitet mit diesem Zahlensystem, das lediglich die Ziffern 0 und 1 kennt.

Bit: Kleinste Informationseinheit für den Computer, auf der die gesamte Datenverarbeitung beruht (8 Bit = 1 Byte = 1 Buchstabe oder Zeichen).

Bug: Fehler im Programm (eigentlich "Wanze", die früher die mechanischen Rechner ab und zu außer Betrieb setzte).

Byte: Entspricht 8 Bit. 1 Byte ist z.B. 1 Buchstabe oder ein Zeichen. Man rechnet beim Computerspeicher mit jeweils 1024 Byte = 1 Kilobyte.

C: schnelle Programmiersprache, die nicht sehr leicht zu erlernen ist, da die Fehlerbehebung erst nach dem Umwandeln in Maschinensprache (Kompilieren) vorgenommen werden kann.

Cartridge: Plastikkästchen, in dem eine Platine steckt, die z.B. ein Programm enthält.

Centronics: Firma, die z.B. den Druckeranschluß für Ihren ST entwickelt hat.

Chip: Elektronischer Baustein, der auf fotographischem Wege hergestellt wird und eine Unmenge von Transistoren enthält.

CP/M: Name eines Betriebssystems, das hauptsächlich für kommerzielle Programme benutzt wird.

CPU: (Central Processing Unit = Zentraleinheit) Herzstück des Computers, das sämtlichen Datenverkehr steuert.

Cursor: Aktuelle Schreibmarke auf dem Bildschirm.

Debuggen: Befreien der Programme von Fehlern (eigentlich: Entfernen der Wanzen=bugs aus dem Computer).

Dezimalsystem: Zahlsystem, mit dem wir rechnen (Ziffern zwischen 0 und 9).

Diskette: Magnetbeschichtete Plastikscheibe zum Speichern von Daten. Maße: 3, 3.5 oder 5.25 Zoll.

Diskettenstation: Gerät zum Abspielen von Disketten.

Drucker: Gerät zum Schreiben von Daten auf Papier.

Dualsystem: (auch Binärsystem genannt) Der Computer arbeitet mit diesem Zahlsystem, das lediglich die Zahlen 0 und 1 kennt.

Editieren: Korrigieren von Fehlern direkt auf dem Bildschirm.

EPROM: Speicherbaustein, auf dem Daten festgehalten werden können=Brennen des EPROMs. Löschen des EPROMs durch UV-Licht.

Errormeldung: Fehlermeldung, die auf dem Bildschirm ausgegeben wird.

Expansionsport: Anschlußbuchse Ihres ST, die zur Erweiterung dient.

Fenster: siehe unter Window

Feuerknopf: Meist auf Joysticks zur Spielsteuerung gedacht.

Floppy Disk: (biegsame Scheibe) Bezeichnung für Diskette bzw. Diskettenstation (vgl. Harddisk = harte Scheibe).

Funktionstasten: Tasten, die von Programmen mit bestimmten Funktionen belegt werden können oder bereits mit einem Kommando belegt sind (leichter zu handhaben, als das Kommando buchstabenweise einzugeben).

GEM: Graphics Environment Manager. Grafikbetriebssystem, das die Arbeit mit bis zu 4 Windows gleichzeitig gestattet.

Grafik: Darstellung von Zeichnungen

Harddisk: Gerät zum sehr schnellen Speichern und Laden großer Datenmengen. (Harddisk = harte Scheibe, vgl. Floppy Disk: biegsame Scheibe).

Hardware: Alle mechanischen und elektronischen Teile des Computers (Gegenteil: Software).

Hexadezimalsystem: Zahlsystem, das aus 16 Ziffern bzw. Zeichen besteht (0 bis 9 und A bis F).

Icon: Bildsymbol, das eine Handlung veranschaulicht (z.B. Programm in Form eines Blocks, Diskettenstation in Form eines Karteikastens).

Interface: Verbindungsstelle zwischen Computer und angeschlossenen Geräten (Peripherie).

Joystick: Steuerinstrument mit frei beweglichem Knüppel und Feuerknöpfen (besonders für Spiele geeignet).

Kilobit: 1024 Bit = 128 Byte; Maßzahl für Speicherbausteine.

Kilobyte: 1024 Byte; Maßeinheit für die Größe des Computerspeichers.

Kompatibilität: Austausch von Hard- bzw. Software zwischen verschiedenen Computern möglich.

Lichtgriffel: Zeichengerät, das mit einer Fotodiode ausgestattet direktes Zeichnen auf dem Bildschirm ermöglicht.

Lightpen: siehe unter Lichtgriffel

LOGO: Programmiersprache, die verhältnismäßig leicht zu erlernen ist (weil sofort Fehler erkannt werden und man sich die Befehle gut merken kann). Grafikorientiert, "Schildkröte".

Malpad: Gerät, das auf einer postkartengroßen Fläche die Erstellung von Grafik ermöglicht (Aufbringen der Grafik mit einem Plastikgriffel).

Maschinensprache: Programmiersprache, mit der Ihr Computer eigentlich arbeitet, nachdem er sich z.B. die BASIC-Befehle in diese Maschinensprache übersetzt hat.

Matrixdrucker: Auch Nadeldrucker genannt, weil er mit Hilfe von Nadeln Buchstaben, Zeichen und Grafiken in Form einer Matrix zu Papier bringt.

Maus: Steuerinstrument, das aus einem kleinen Kästchen mit innenliegender Kugel besteht (bei Bewegung dieses Kästchens auf dem Schreibtisch bewegt sich der Cursor auf dem Bildschirm).

Mega: Bezeichnung für ca. 1 Million.

Mikroprozessor: siehe unter CPU

Modulator: Durch dieses Bauteil wird das Computerbild in ein auf dem Fernseher darstellbares Signal umgewandelt.

Monitor: Spezielles Datensichtgerät. Bietet durch höhere Bildwiederholung eine flimmerfreie Darstellung.

MSX: Abkürzung für Microsoft Super Extended: Bezeichnung für BASIC und Betriebssystem der untereinander kompatiblen MSX-Computer.

Oktalsystem: Zahlsystem, das aus 8 Ziffern besteht (0 bis 7).

Pad: siehe unter Malpad

Paddle: Steuerinstrument, das über einen Drehknopf und meist einen Feuerknopf verfügt.

Parallel-Anschluß: Hier werden gleichzeitig 8 Bit übertragen (Gegenteil: seriell=gleichzeitig 1 Bit).

Peripherie: Alle Geräte, die zusätzlich mit dem Computer verbunden werden können (z.B. Drucker, Floppy).

Programmiersprache: Anzahl von Befehlen, die es ermöglichen, den Computer zu steuern. Es gibt die sehr schwierig zu erlernende, computereigene Maschinensprache und höhere Programmiersprachen wie BASIC und LOGO.

PROM: Speicherbaustein, auf dem Daten festgehalten werden können. Das Löschen des PROMs ist nicht möglich.

Pull-Down-Menü: Hinter einem Stichwort in der obersten Bildschirmzeile verbirgt sich ein Menü, das nach dem Berühren mit dem Mauspfeil nach unten ausgerollt wird.

RAM: Speicherbaustein, auf dem Daten festgehalten werden können. Löschen des RAMs, wenn kein Stromzufluß mehr besteht.

Relais: Elektromechanischer Schalter. Später ersetzt durch Transistoren und schließlich Chips.

RGB: Abk. für Rot-Grün-Blau. Kennzeichen guter Farbmonitoren.

ROM: Nur-Lese-Speicher, der auch bei Stromausfall nicht seine Daten verliert (Unterscheidung EPROM, PROM und RAM).

RS-232-Schnittstelle: Verbindung zu anderen Geräten; die Datenübertragung erfolgt über diese Schnittstelle bitweise.

Schreibmarke: Aktueller Cursor auf dem Bildschirm.

Serielle Datenübertragung: Die Datenübertragung erfolgt bitweise (Gegenteil parallel = 8 Bit).

Software: z.B. Programmiersprachen, Betriebssysteme und fertige Programme.

Sound: Klänge oder Geräusche, die vom Computer mit einem. speziellen Chip erzeugt werden

Speicher: Elektronische Bausteine, die Daten festhalten können: RAM, ROM, EPROM, PROM.

Steuercode: Durch Tastendruck kann z.B. der Bildschirm gelöscht werden.

String: Schublade bzw. Variable, in der Texte abgespeichert werden können.

Synthesizer: Elektronisches Bauteil, das künstlich erzeugte Töne und Geräusche wiedergeben kann.

TOS: Tramiel Operating System. Betriebssystem Ihres ST.

Transistor: Elektronisches Schaltelement (vorher Relais), später ersetzt durch Chip (mehrere tausend Transistoren auf der Größe eines Stecknadelkopfes).

Variable: Teil des Speichers (Schublade), in dem bestimmte Daten enthalten sind.

Window: Einer von mehreren Bildschirmen, die zur gleichen Zeit dargestellt werden können und sich eventuell z.T. überlagern.

Zeichensatz: Satz von Buchstaben und anderen Zeichen, die unserer Computer von vornherein beherrscht.

Zentraleinheit: siehe unter CPU

68000: Mikroprozessor bzw. CPU bzw. Zentraleinheit, die im ST-Computer enthalten ist.

5.4 Fehlermeldungen des ST-BASIC

Nachfolgend die wichtigsten Fehlermeldungen und ihre Übersetzung, in Klammern stehen die Fehlermeldungen des neuen BASIC.

1: Undefined error (Undefined Error)

Ein nicht definierbarer Fehler ist aufgetreten. Zwar verfügt das ST-BASIC über eine Vielzahl von Fehlermeldungen, aber ab und zu kann auch mal ein Fehler auftreten, mit dem der Computer nichts anzufangen weiß.

2: Something is wrong (Syntax Error)

Irgendetwas ist falsch. Diese Fehlermeldung wird meist dann ausgegeben, wenn Sie eine BASIC- Vokabel falsch geschrieben haben z.B. statt 'PRINT' 'PINT'.

3: RETURN statement needs matching GOSUB (Return without GOSUB)

Wer RETURN sagt, muß vorher auch GOSUB ... gesagt haben. Sie können aus einem Unterprogramm nicht mit RETURN ins Hauptprogramm zurückspringen, wenn Sie das Hauptprogramm vorher nicht mit GOSUB ... verlassen haben.

6: Number too large (Overflow)

Die eingegebene Zahl ist zu groß. Das ST-BASIC verkraftet 'nur' Zahlen zwischen 1E-19 und 1E+18.

7: Not enough memory (Out of memory)

Nicht genug Speicherplatz mehr frei. Je nachdem, ob Sie einen ST mit 512 KByte oder gar einen mit 1 MByte RAM-Speicher besitzen, kann manchmal der Speicherplatz für ein eingegebenes Programm nicht reichen. Jedoch liegt dies zumeist nicht in der Länge des Programmes begründet, sondern vielmehr in zu vielen Variablen, die Sie verwendet haben.

11: You cannot divide by zero (Division by zero)

Sie können eine Zahl nicht durch 0 teilen ... eine alt bekannte Rechenregel, die wohl keiner Erläuterung bedarf.

13: Types of value do not match (Type mismatch)

Die Zuweisung zur Variablen paßt nicht. Sie können einer Zahlvariablen wie z.B. 'A' keinen String zuweisen. Für diesen Zweck gibt es spezielle Stringvariablen wie z.B. A$.

15: Strings cannot be over 255 characters long (String too long)

Strings dürfen nicht länger als 255 Zeichen lang sein. Wollen Sie einen längeren Text in Strings abspeichern, müssen Sie sich mehrerer Stringvariablen bedienen.

17: CONT works only in BREAK mode (Cont valid only in BREAK mode)

Der Befehl CONT(inue) = das Programm weiterlaufen lassen, wirkt nur dann, wenn das Programm mit BREAK unterbrochen wurde (z.B. durch gleichzeitiges Drücken der Tasten 'Control' und 'G') oder der Befehl END im Programm auftauchte

23: Program line too long (Program line too long)

Diese Programmzeile ist zu lang. Eine Programmzeile im ST-BASIC darf höchstens 255 Zeichen lang sein.

30: Window number invalid (Undefined Error)

Diese Fensternummer gibt es nicht. Sie können mit den Fenster-Befehlen nur die Nummern 0 bis 3 ansprechen (0 = EDIT, 1 = LIST, 2 = OUTPUT, 3 = COMMAND).

53: File not found on disk drive specified (File not found)

Auf der gerade im aktuellen Laufwerk befindlichen Diskette befindet sich das gewünschte Programm nicht. Wenn Sie von der Diskette mit LOAD "Name" ein Programm laden wollen, sollten Sie sich vorher das Inhaltsverzeichnis ansehen.

58: File exists (File exists)

Diesen Programmnamen gibt es schon auf der aktuellen Diskette. Wollen Sie ein Programm mit gleichem Namen wieder abspeichern, ist das nicht problematisch, es wird durch das neue Programm ersetzt. Nur dann, wenn ein anderes Programm mit dem gleichen Namen abgespeichert werden soll, wie er bereits existiert, führt dies aus Sicherheitsgründen zu Problemen. In diesem Fall müssen Sie das alte Programm vorher erst von der Diskette löschen.

61: Disk full (Disk full)

Die Diskette ist voll. Sie müssen sich entwederdazu entschließen, einige Programme auf der Diskette zu löschen oder eine neue Diskette nehmen.

64: Invalid filename (Invalid file name)

Der gewählte Programmname ist unzulässig.Ein Programmname darf aus bis zu 8 Zeichen und einem 3-Zeichen-Kennzecihner bestehen.

99: -- Break -- (-- Break --)

Programmunterbrechung. Dieses Signal wird dann auf dem Bildschirm ausgegeben, wenn Sie das laufende Programmz.B. mit 'Control'-'G' unterbrechen. Sie können anschließend mit'CONT' den Programmablauf fortsetzen.

103: Invalid line number (Invalid line number)

Die eingegebene Zeilennummer ist unzulässig. Zeilennummern dürfen nicht kleiner als 0 und nicht größer als 65529 sein.

106: Line number does not exist (Statement not found)

Diese Zeilennummer existiert nicht. Sie können mit GOSUB nicht in ein Unterprogramm bzw. zu einer Zeilennummer springen, die es nicht gibt.

107: Number too large for an integer

Für eine ganze Zahl ist der eingegebene Zahlwert zu groß. Integerzahlen dürfen nur zwischen -32767 und 32767 beim alten und zwischen -2^{31} und 2^{31} beim neuen BASIC groß sein..

108: Input data is not valid, restart input from first item (Redo from start)

Sie dürfen bei einem INPUT-Befehl, wenn eine Zahlvariable eingegeben werden soll, nicht einen String eingeben. Ansonsten müssen Sie die INPUT-Eingabe noch einmal von vorne beginnen.

109: STOP (STOP)

Programmunterbrechung. Sie können das Programm nur durch 'RUN' neu starten.

204: FOR statement needs a NEXT or WHILE needs a WEND (FOR without NEXT)

Zu einer 'FOR...NEXT'-Schleife gehört ein FOR und ein NEXT, sonst kann die Programmschleife nicht einwandfrei durchgeführt werden. Zu einer 'WHILE...WEND'-Schleife gehört ein WHILE und ein WEND, sonst kann die Programmschleife nicht einwandfrei durchgeführt werden.

205: NEXT statement needs a FOR or WEND needs a WHILE (NEXT without FOR)

Zu einer 'FOR...NEXT'-Schleife gehört ein FOR und ein NEXT, sonst kann die Programmschleife nicht einwandfrei durchgeführt werden. Zu einer 'WHILE...WEND'-Schleife gehört ein WHILE und ein WEND, sonst kann die Programmschleife nicht einwandfrei durchgeführt werden.

221: System error ..., please restart (System error #%u)

Systemfehler, bitte das Programm neu starten.

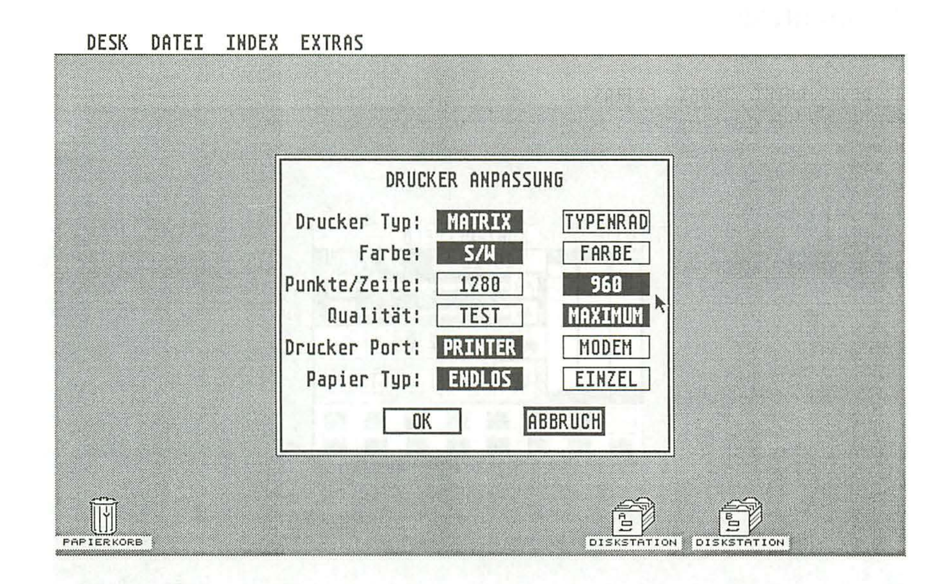
223: Too many FOR loops (Too many FOR loops)

Man darf nicht beliebig viele FOR...NEXT-Schleifen ineinander schachteln.

5.5 Druckeranpassung und Kontrollfeld

Drucker Anpassung

Die Druckeranpassung, die beim ersten Pull-Down-Menü zur Auswahl steht, sieht folgendermaßen aus:

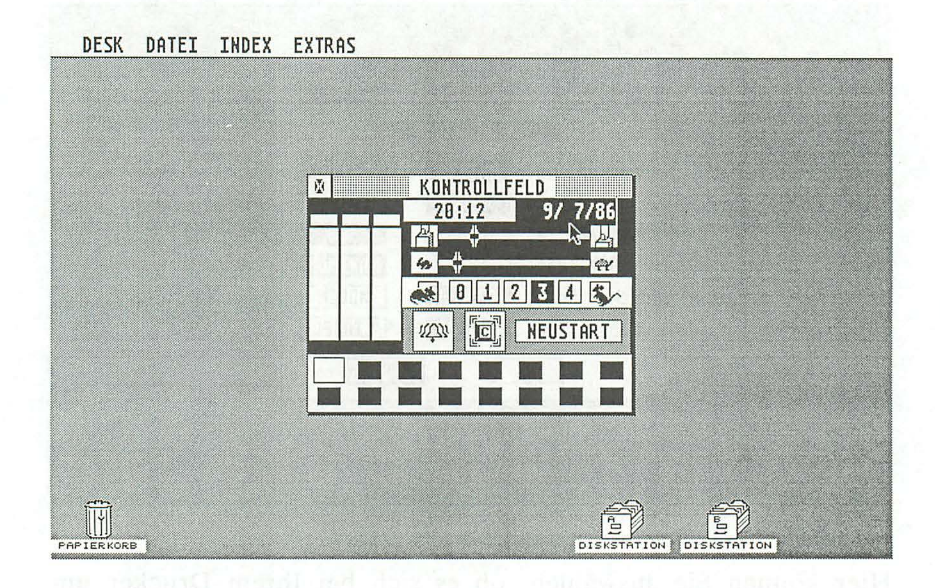


Hier können Sie auswählen, ob es sich bei Ihrem Drucker um einen Matrix- oder einen Typenraddrucker, einen Schwarz/Weiß- oder aber gar einen Farbdrucker handelt. Sehr wichtig ist es zudem, bei einem EPSON- oder einem EPSON-kompatiblen Drucker vor dem Bildschirmausdruck (mit Hardcopy) die Punkte pro Zeile auf 960 und die Druckqualität auf MAXIMUM einzustellen; ansonsten gibt Ihr ST nur einen Teil des Bildschirms auf den Drucker aus. Besitzen Sie einen Drucker mit RS232-Port, sollten Sie in der Druckeranpassung den Parameter MODEM mit der Maus anklicken, bei einem Centronics-Drucker hingegen wählen Sie bitte die Einstellung PRINTER. Schließlich können Sie noch einstellen, ob mit Einzelblättern

oder mit Endlospapier gearbeitet werden soll. Diese Einstellung funktioniert aber bisher nur in Zusammenarbeit mit bestimmten Programmen.

Kontrollfeld

Ebenfalls im ersten Pull-Down-Menü finden wird folgendes Kontrollfeld:



Je nach Feld, das Sie mit dem Pfeil ansteuern und mit der Maustaste bestätigen, können Sie hier zwischen folgenden Funktionen wählen: Uhrzeit bzw. Datum, Schnelligkeit vom Tastenandruck bis zur ersten bzw. bis zur Dauerwiederholungsrate, Tastaturklick, Fehlersignal, Farbe und Maus-Doppelklickgeschwindigkeit.

Zur Farbeinstellung (falls Sie stolzer Besitzer eines an Ihren ST passenden Farbmonitors sind) können Sie auch die drei Schieberegler auf der linken Seite des Kontrollfeldes (für Rot-, Grün- und Blau-Anteil) ansteuern.

Als Besitzer eines Schwarz/Weiß-Monitors mit höchster Auflösungsstufe (640*400 Punkte) haben Sie jedoch nur die Möglichkeit, zwischen weißer Bildschirmseite und schwarzer Schrift oder dem Gegensatz, schwarzer Bildschirmseite und weißer Schrift, auszuwählen. Sie brauchen zu diesem Zweck mit der Maus lediglich die drei Schieberegler in die entgegengesetzte Richtung zu schieben (bis zum Anschlag), schon sind alle zukünftigen Aktionen im Aussehen umgekehrt von denen, wie sie bisher auf dem Bildschirm abliefen.

Noch ein paar Erklärungen zu den verschiedenen Funktionen des Kontrollfeldes:

Uhrfunktion: Sie klicken das Uhrfeld an, löschen die alte Anzeige mit der Backspace-Taste (oder drücken einmal die Esc-Taste), und geben nun die aktuelle Uhrzeit ein. Sie bestätigen diese Aktion durch abermaliges Anklicken des Uhrfeldes mit der linken Maustaste oder der RETURN-Taste.

Datumsfunktion: Hier verfahren Sie genauso wie bei der Uhrfunktion. Denken Sie aber bitte daran, sowohl bei der Uhrzeit als auch beim Datum keine ungültigen Eingaben vorzunehmen. Ihr ST akzeptiert diese nämlich nicht, sondern zeigt in diesem Fall ein anderes, ähnliches Datum (das es aber auch wirklich gibt) an.

Einstellung des ersten wiederholten Tastaturanschlags: Bewegen Sie den Zeiger in dieser Zeile an die Stelle, an der es Ihnen beliebt. Befindet sich der Schalter sehr weit links, setzt die Wiederholung des Tastaturanschlags sofort beim Tastenandruck ein, haben Sie den Schalter ganz nach rechts gestellt, dauert es bis zum ersten wiederholten Tastenanschlag länger als eine Sekunde.

Einstellung der Geschwindigkeit beim weiteren Wiederholen des Tastenanschlags: Hier sprechen die Bilder wohl für sich. Befindet sich der Zeiger nahe dem Hasen (linke Seite), geht die Wiederholung äußerst schnell vonstatten; befindet sich der

Zeiger hingegen nahe der Schildkröte, dauert es Ewigkeiten (länger als eine Sekunde), bis eine Taste beim Niederdrücken wiederholt wird.

Einstellung des Doppelklicks bei der Maus: Auch hier sprechen die Symbole für sich: Stellen Sie die Zahl 0 ein (ruhende Maus auf der linken Seite), können Sie den Doppelklick recht langsam durchführen, Ihr ST versteht ihn. Haben Sie jedoch Position 4 angewählt (aufgerichtete Maus auf der rechten Seite), müssen Sie sich schon beeilen, um einen solch schnellen Doppelklick ausführen zu können. Als ideal haben sich die Einstellungen 2 und 3 herausgestellt.

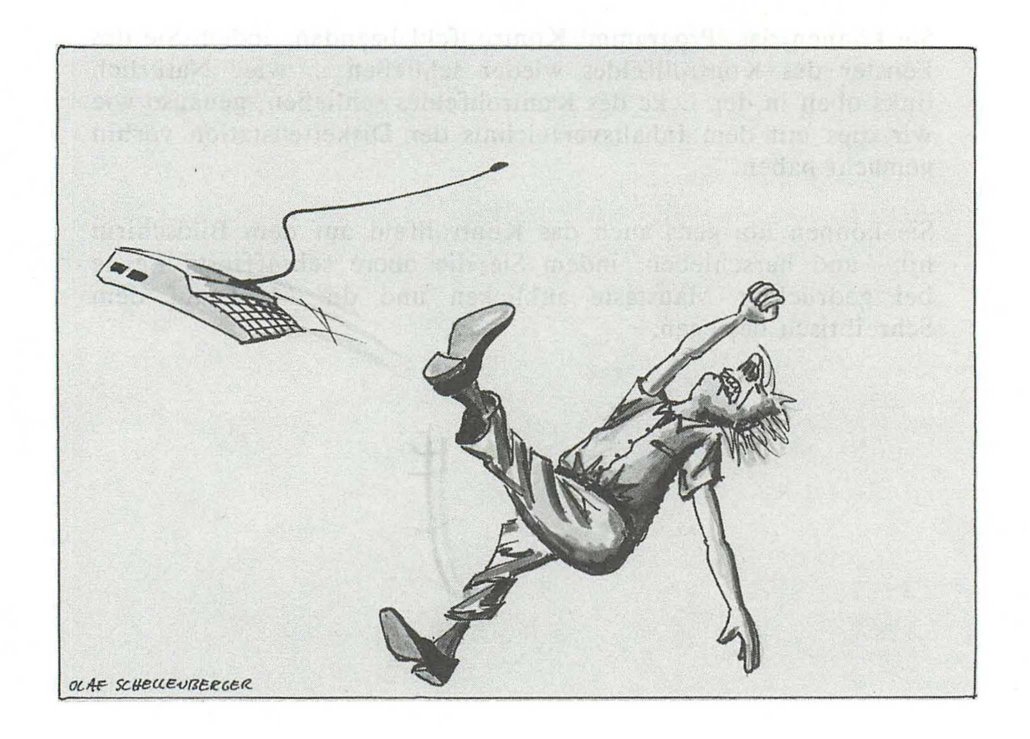
Glocke und Tastenklick: Die zwei folgenden Symbole können aus- oder eingeschaltet werden, indem Sie den Mausknopf an der entsprechenden Stelle ein- oder zweimal betätigen. Die Glocke erklingt, wenn Sie einen gravierenden Fehler begangen haben, der Tastaturklick ertönt, wenn Sie eine Taste betätigen (als Kontrolle sind beide Funktionen sehr wertvoll). Anstatt über das Kontrollfeld eine dieser beiden Funktionen an- bzw. auszustellen, raten wir Ihnen, bei Bedarf (falls jemand anders während Ihrer Arbeit am Computer durch das andauernde Piepsen gestört werden sollte) den Lautstärkeregler am Monitor soweit möglich zurückzudrehen.

Die drei Schieberegler für RGB wurden schon erwähnt: Sie können hiermit bis zu 16 Farben im unteren Teil des Kontrollfeldes abspeichern und dieselbigen auch wieder durch Ansteuern mit der Maus im Bedarfsfall anwählen (bei Schwarz/Weiß-Monitoren haben diese 16 Einstellungen keine Funktion).

Schließlich noch die NEUSTART-Taste im Kontrollfeld: Gefällt Ihnen Ihre veränderte Eingabe nicht, können Sie die Neustart-Taste betätigen. Augenblicklich werden alle veränderten Werte auf ihren ursprünglichen Wert zurückgesetzt, nur Datum und Uhrzeit bleiben auch nach dem NEUSTART erhalten.

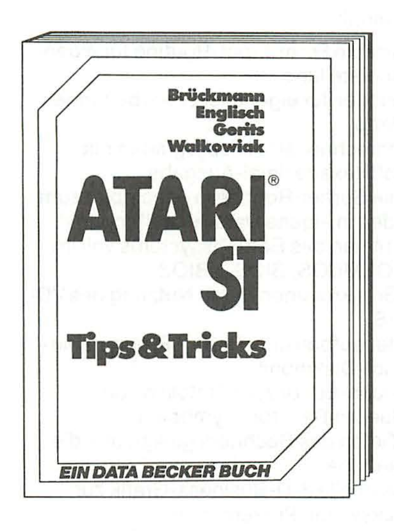
Sie können das 'Programm' Kontrollfeld beenden, indem Sie das Fenster des Kontrollfeldes wieder schließen ... wie? Natürlich links oben in der Ecke des Kontrollfeldes schließen, genauso wie wir dies mit dem Inhaltsverzeichnis der Diskettenstation vorhin gemacht haben.

Sie können übrigens auch das Kontrollfeld auf dem Bildschirm hin- und herschieben, indem Sie die obere schraffierte Leiste bei gedrückter Maustaste anklicken und die Maus auf dem Schreibtisch bewegen.


OLAF SCHELLENBERGER

5.6 Stichwortverzeichnis

Eine riesige Fundgrube wirkungsvoller Tips & Tricks rund um den neuen ATARI ST. Alle Programme sind gut erklärt und können in eigene Anwendungen eingebaut werden. Diese Routinen sind wirklich absolut neu.



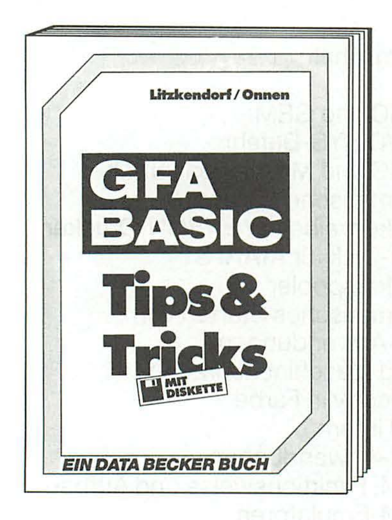
Aus dem Inhalt:

- BASIC und GEM
- Der VDISYS-Befehl
- BASIC und Maschinensprache
- Automatische Hardcopy
- Druckertreiber für EPSON-Drucker
- RAM-Disk für ATARI ST
- Druckerspooler
- Automatisches Starten von TOS-Anwendungen
- C und Maschinensprache
- Hardcopy in Farbe
- GEM intern
- GEM-Anwendungen
- CP/M, Funktionsweise und Aufbau
- CP/M-Emulatoren

und vieles mehr

Brückmann, Englisch, Gerits, Walkowiak
ATARI ST Tips & Tricks
Hardcover, 362 Seiten, DM 49,–
ISBN 3-89011-118-1

GFA-BASIC ist ohne Zweifel eine der leistungsfähigsten BASIC-Versionen, die es für den ATARI ST gibt. Nur – wer diese fantastischen Fähigkeiten wirklich voll ausschöpfen will, braucht entsprechendes Know-how, braucht bei der Programmierung all die hilfreichen Tips & Tricks der GFA-Profis. Und damit der Spaß nicht beim Abtippen aufhört, werden alle Beispielprogramme gleich auf Diskette mitgeliefert.



Aus dem Inhalt:

- Komfortable Form-Input-Routine für eigene Maskenerstellung
- Hilfsroutinen für eigene Textverarbeitungen – WordWrap
- Taschenrechner als Hilfsprogramm mit Dezimal/Hexadezimal-Ausgabe
- Schnelle Sortier-Routine in Assembler zum Einbinden in eigene Dateiverwaltungen
- Die Routinen des Betriebssystems voll im Griff – GEMDOS, BIOS, XBIOS
- Neue Grafikroutinen durch Nutzung des VDI und AES
- Diskettenaufbau und Floppy-Programmierung, incl. Diskmonitor
- Super-Icon-Editor zum Erstellen von individuellen Desktop-Symbolen
- In die Tiefen des Rechners geschaut – die Speicherlupe
- Professionelle 3-D-Business-Grafik zur eindrucksvollen Präsentation
- GEM-Programmierung mit GFA-BASIC: die eigene Benutzeroberfläche, Pull-Down-Menüs, Windows, Eingabeboxen und beliebige Grafiken
- Das schier Unmögliche – 7 Fenster gleichzeitig öffnen, ohne daß es zieht
- Resource Construction Set in GFA-BASIC als Baukasten

Litzkendorf, Onnen
GFA-BASIC Tips & Tricks
350 Seiten, DM 49,–
ISBN 3-89011-193-9

Bücher zum ATARI ST

Verlassen Sie sich nicht auf Gerüchte, profitieren Sie lieber vom DATA BECKER-Know-How. Keine nackten Befehlsübersichten, sondern wirklich brauchbares Material liefert dieses Buch in Hülle und Fülle. Dem Autor ist es erstmalig gelungen, ein wirklich spannendes Buch zum neuen GfA-BASIC zu schreiben. So lernen Sie nicht nur schnell und kompetent das GfA-BASIC und die Systemprogrammierung des ATARI ST kennen, sondern erhalten zusätzlich noch ein komplettes GRAPHIC CONSTRUCTION SET, das mit den gebotenen Features garantiert einmalig ist.



Aus dem Inhalt:

- Kommandos des GfA-EDITORS
- Strukturierte Programmierung kontra Spaghetti-Code
- Professionelle Programmerstellung anhand des GCS
- Trickfilm-Modus in Echtzeit
- Patch-Grafik-Design
- 3-D-Animationsgrafik
- Pattern- und Grafik-Editor
- Verzerren, Spiegeln, 3-D-Trommel-Generator
- Clipping, Blockverarbeitung, Netzmodus
- 2 aktive/2 passive Grafikseiten und vieles mehr

Litzkendorf
Das große GfA-BASIC-Buch
Hardcover, 468 Seiten, DM 49,–
ISBN 3-89011-222-6

Keine Programmiersprache für den ATARI ST ist zur Entwicklung neuer Programmprojekte so gut geeignet wie C. Schließlich wurde die Benutzeroberfläche GEM fast vollständig in dieser Sprache geschrieben. Doch dies ist nicht der einzige Grund, warum viele Profis und Softwarehäuser ihre Programme vorzugsweise unter C erstellen. C ist trotzdem protabel, klar strukturiert und modular. Wer nun diese starke Sprache für seine eigenen Programme nutzen will, findet hier die richtige Einführung. Mit vielen Beispielen, Tips und Tricks formt dieses Buch jeden Einsteiger zum C-Spezialisten.



Aus dem Inhalt:

- Einführung in C
- Variablen und Arithmetik in C
- Funktionen und Parameter
- Logische Verknüpfungen
- Speicherklassen
- C-Features
- Datenfelder
- Zeigerarrays
- Grundlagen der GEM-Programmierung
- Window-Programmierung
- Zahlreiche Tips & Tricks
- Beschreibung des Data Becker C-Compilers

Schaun
Atari ST – C für Einsteiger
393 Seiten, DM 39,–
ISBN 3-89011-183-1

Bücher zum Atari ST

Der Atari ST ist durch seinen guten Monitor, die hohe Auflösung und die vorhandene Speicherkapazität sowohl für Textverarbeitung als auch für Graphik hervorragend geeignet. Mit 1st Word Plus werden diese Komponenten in sinnvoller Weise vereint. Zusammen mit dem Erweiterungsprogramm in 1st Word Plus entsteht somit ein sehr komfortables Textverarbeitungssystem, welches selbst hohen Ansprüchen gerecht wird, wie die vielen Abbildungen in diesem Buch beweisen.

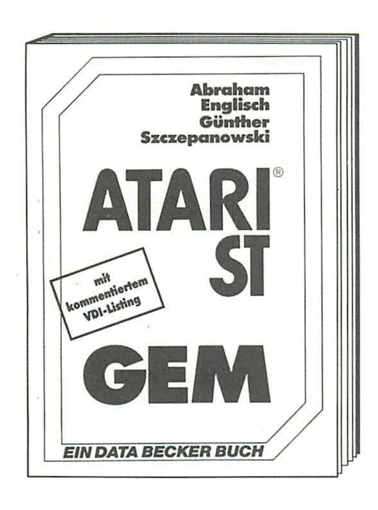


Aus dem Inhalt:

- 1st Word Plus
- 1st Mail
- 1st Lektor
- 1st Proportional
- 1st Index
- Textgestaltung mit 1st Word Plus
- Wörterbücher
- Zusammenarbeit der verschiedenen Programme
- Umfangreiches Kapitel über Druckeranpassung
- Durch neue Druckertreiber neue Möglichkeiten und verbesserte Leistung
- Graphik im Text
- Desktop Publishing mit 1st Word Plus?
- Ausführliches Glossar

Kraus
Das große Buch zu 1st Word Plus
288 Seiten, inkl. Diskette, DM 59,–
ISBN 3-89011-190-4

Ein Buch für jeden, der das Betriebssystem der Zukunft verstehen, anwenden und die gigantische GEM-Bibliothek nutzen will! Grundlegende Informationen wie die Organisation des GEM im ATARI ST, die verwendbaren Programmiersprachen, die Funktionen des Virtual Device Interface und des Application Evironment System sind ausführlich und detailliert erklärt.



Aus dem Inhalt:

- GEM-Programmierung in Assembler, C und GfA-BASIC
- Objektbäume und Formulare in GfA-BASIC
- RCS-Bedienung
- Programmierung von Slider-Objekten
- Erstellen eigener GEM-Bindings
- Dokumentation sämtlicher AES-Funktionen
- Listing ausgewählter AES-Routinen
- Kommentiertes VDI-ROM-Listing

Abraham, Englisch, Günther, Szczepanowski
Atari ST GEM
Hardcover, 691 Seiten, DM 69,–
ISBN 3-89011-251-X



DAS STEHT DRIN:

Geht es Ihnen auch so? Da hat man sich ein neues Gerät gekauft und hofft, mit Hilfe der mitgelieferten Gebrauchsanleitung nun vom Anfänger zum Profi aufzusteigen. Von ersten Schwierigkeiten jäh gebremst, sucht man dann im Karton nach der Gebrauchtsanleitung zur Gebrauchsanleitung ...
Dieses Buch zeigt, wie Sie ohne derartige Umwege Ihren ATARI ST kennenlernen und ihn beherrschen. Nebenbei bekommen Sie eine Menge Computer-Grundwissen vermittelt. Schritt für Schritt werden u. a. behandelt:

Aus dem Inhalt:

- Aufstellen und Anschließen
- Das Betriebssystem TOS mit GEM und Anwendungen
- Disketten kopieren
- Einführung ins Programmieren in BASIC
- Erste eigene Programme
- Einige fertige Programme zum Abtippen mit ausführlicher Erläuterung
- Grundbegriffe der Computerei
- Anschlußmöglichkeiten
- Ein Häppchen Computergeschichte
- Was man noch so alles mit seinem ATARI ST anstellen kann

UND GESCHRIEBEN HABEN DIESES BUCH:

Rainer Lüers ist ausgebildeter Lehrer, Computer-Fachtrainer und erfolgreicher Autor verschiedener DATA BECKER-Bücher. Michael Stein, Redakteur der DATA WELT, machte das Buch durch seine fundierten Computerkenntnisse und seinen lockeren, humorvollen Stil zu einem angenehm lesbaren, leichtverständlichen Buch für jedermann.

